Improving Engineering Efficiency with PLM/ALM

Dr. Christof **Ebert**, Vector, Germany

Source and copyrights:

Ebert, C.: Improving engineering efficiency with PLM/ALM, Software & Systems Modeling, July 2013, Volume 12, Issue 3, pp 443-449.
www.vector.com/portal/medien/vector_consulting/publications/Ebert_ImprovingEngineeringEfficiencyWithPLMALM.pdf

Abstract

Rising cost pressure is forcing manufacturers and their suppliers to jointly and consistently master product development. Our industry case study shows how a leading automotive OEM over time has achieved effective interaction of engineering processes, tools and people on the basis of product and application life-cycle management (PLM / ALM).

Its scope is first an introduction to PLM/ALM on the basis of a model driven engineering (MDE) for one or several products or product families. Second, PLM and ALM need tool support to the degree necessary to ease handling and drive reuse and consistency. Third, introducing MDE needs profound change management.

Starting from establishing the relevant engineering processes, we show how they can be effectively automated for best possible usage across the enterprise and even for suppliers. We practically describe how such a profound change process is successfully managed together with impacted engineers and how the concepts can be transferred to other companies. Concrete results for efficiency improvement, shorter lead time and better quality in product development combined with better global engineering underline the business value.

Keywords

ALM, Application Lifecycle Management, PLM, Product Lifecycle Management, Efficiency, Industry Voice

1. Improving Efficiency with Better Engineering Processes

Companies and entire industries are changing very fast at this time. Hardly any year has changed our lives with technological breakthroughs as strong as the last twelve months have done. With the explosion in Fukushima, the energy change has been started in an unprecedented push of industry and politics. The high-speed train accident in China emphasized that functional safety and quality must be uncompromised. Space Shuttle had its last flight and showed us that the complexity of technical systems must remain both technically and economically manageable. The first electric cars came into series production and showed the feasibility of entirely new powertrain technologies. Smartphones already make half of all cell phone sales.This all is the basis for many innovations in mobility, which will continue to thrive in 2012.

Software and IT are the main drivers of innovation. Software-intensive systems as used in automobiles, aircraft, medical, transportation, utilities and industrial automation deliver today 50-70 percent of the value of these solutions, and this will further grow. The products and solutions must meet increasing quality requirements, but must be developed cost-efficiently, should be easily adaptable to new environments and must effectively exploit the advantages of modern technologies. At the same time new competitors are pushing new solutions on the market, and the technology landscape is increasingly cluttered.

This economic climate enforces two trends in engineering across industries, namely fast innovation and also cost efficiency. Companies invest in growth through innovation by developing new products and solutions. At the same time they are aware of the volatile market situation and thus challenge their development teams worldwide to be as lean and efficient as possible. A senior R&D manager of a leading automotive tier 1 supplier summarized it as follows: “Cars sell in quantities we have never seen. Markets around the world are keen to get latest technology with their cars such as energy efficiency, functional safety or internet access. But we have learned our lessons from the recent recession and made engineering and production flexible to immediately react if sales drop.”

There are numerous levers for engineering efficiency improvement. Many companies operate with distributed teams leading to fragmented processes and tool chains with heterogeneous interfaces, redundant and inconsistent data management and insufficient transparency which results have been achieved and what needs still to be done. The underlying root cause is the lack of a shared conceptual model of the product. In consequence, activities such as project management, pre-development and product engineering are rarely integrated well due to the diversity of stakeholders with individual knowledge about projects, products and processes. As a result, engineering results such as specifications, documentation and test cases are inconsistent, items like signals and parameters are arbitrarily labeled, changes create lots of extra work to make sure that nothing is overlooked, and reuse is hardly possible due to the many heterogeneous contents. This pattern is amplified when collaboration across supplier networks comes into the picture, as it is today normal in systems development.

A brief example shows the significance. A supplier is introducing MDE based on some methodology, processes and a modern tool environment that enables seamless collaboration across development centers and with partners and customers. In advance, cost-effectiveness was evident because the system was going to provide faster access to data and less defects and budget overruns due to improved change and configuration control. After the introduction phase however, a MDE tools environment was available, but did not deliver useful models. Engineers were still drawing their previous style pictures, without much modeling methodology. What happened? The tool was designed to support development and was integrated into the company-wide PDM system. Electronic developers but also product managers and project leaders could not work with it and created parallel systems for their documents, which they exchanged between one another using traditional methods. The solution would have been simple if it had been made clear, before introducing the new tool, which processes had to be supported and how these processes had to be first improved and then automated.

An objective-driven tool introduction, which achieves measureable improvements and is accepted by engineers, requires good preparation and implementation. Normally, multiple departments or entire divisions are affected by a change of engineering and tools processes. Before debating tools solutions, the processes have to be under control and the envisaged workflows and work organization must be understood and optimized with the impacted engineers. An efficient electronics development builds upon streamlined processes that are understood and practiced by all engineers, because workflows are well orchestrated and optimally support engineering tasks. Success with ALM and PLM implies that processes and tools are simultaneously improved.

We will show with this article how engineering processes can be improved and automated, thus enhancing efficiency, quality and lead time. Such changes need leadership and good orchestration to be successful. We therefore show how a sustainable change process is successfully managed together with impacted engineers and how the concepts can be transferred to other companies.

2. Product Life-Cycle Management

Product Life-Cycle Management (PLM) and its equivalent in software, namely Application life-cycle management (ALM), is the overall business process that governs a product or service from its inception to the end of its life in order to achieve the best possible value for the business of the enterprise and its customers and partners. PLM/ALM combines processes, people and tools for the effective engineering of products – from their inception until end of service. It involves tacit knowledge of experts and explicit knowledge, codified in procedures, process and tools. PLM/ALM stretches from know-how to know-what and know-why.

There is a huge misconception that PLM/ALM is a product not a process. By definition, it is the process by which organizations manage the creation, deployment, and operation of software over its full lifecycle. In practice, PLM/ALM has been associated with tooling suites aimed at managing the tasks of this lifecycle, but vendors have rarely delivered on the promise of integrating the management of the full application lifecycle. PLM/ALM is only cost-effective if there are the right engineering processes and if they are automated and instrumented with appropriate tools [1,2]. PLM/ALM integration has been a problem in the industry for a long time. On the one hand, customers generally do not replace their favorite point solutions, such as Jazz and TeamCenter, requirements management with PTC Integrity, development tools like Matlab/Simulink or Rhapsody, and replace them with a single suite from one vendor. Even where the PLM/ALM part is replaced with a single suite, there are often other tools to integrate with, such as defect tracking, change management, document management, etc.

To work efficiently, engineers need to handle a multitude of processes and different forms of knowledge to be shared with colleagues across business processes and even beyond the borders of the enterprise [1]. PLM/ALM helps to integrate those along the entire life-cycle of a release or product or beyond to an entire portfolio as is illustrated in Figure 1. Many companies have realized in this fierce climate that their traditional rather organically grown tools landscape with isolated unconnected processes not only won't scale up but also limit their engineering productivity due to manual data exchange, too much rework, inconsistencies and insufficient reuse across products and platforms (Figure 1, left side). A federation of processes and supporting tools with clear responsibilities improves efficiency by more consistency, quality, reuse and not the least employee motivation (Figure 1, right side).

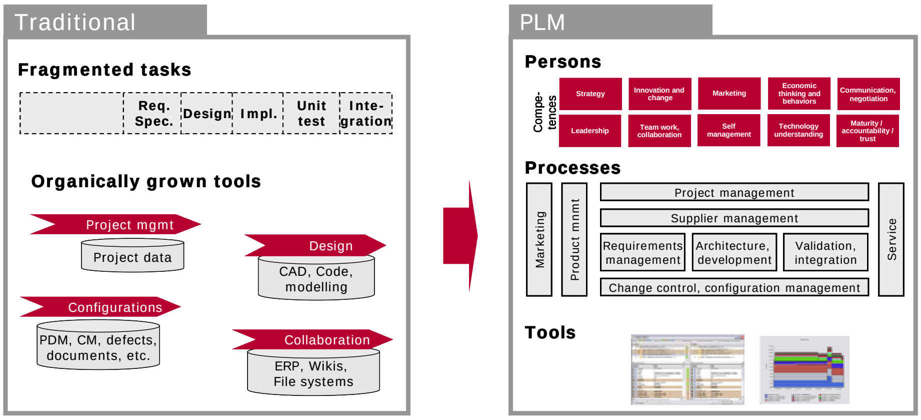

Figure 1: PLM integrates processes, people and tools for effective engineering

PLM/ALM needs both process and tools support. Figure 2 shows this relationship based on a study at London Business School [1,2]. The horizontal axis portrays the degree of tools support in an engineering environment, while the vertical axis shows to which degree the processes are optimized. All four combinations of high versus low have an associated impact on engineering productivity. Obviously the upper right quadrant shows best performance impact, namely 20%. Introducing the right process first and emphasizing its understanding in engineering has a higher value (upper left field), than pushing an engineering tool without profound process understanding (lower right field). Note that the values of course depend on the specific scenario and environment. What matters here is the ratio between the four fields.

Tuning processes, improving project management, and establishing visibility on new product introduction – techniques well described by common improvement frameworks, such as CMMI [4] – will not yield sustainable benefits if not adequately supported by tools. The prospect of new, high-margin products, combined with the delayed impacts of resource allocation decisions, seduce product managers into starting more projects than their development resources can handle.

The scope of a PLM/ALM system is on the one hand the creation and management of engineering data in one common engineering data backbone and on the other hand the management of processes. PLM/ALM as a concept and solution applies to software engineering as well as to systems or hardware products. It applies to different types and sizes of companies, because it is not prescribing a solution suitable only for big companies but rather a clear focus on processes along the life-cycle. We use it for complex solutions with multiple hardware and software components as well as simple software services.

Today such environments need to satisfy a variety of needs for integrated systems engineering, namely

- Integrated business logic and one comprehensive data model for the entire E/E development process from requirements, through system, SW- and HW-design to test.
- Architecture Design and Management
- Collaboration environment for requirements and test Engineering, modeling of functions, components, networks and communication

- Variant management and product line engineering for effective reuse
- Support for quality requirements such as functional safety from requirements to concept, models, realization to validation, proof and acceptance

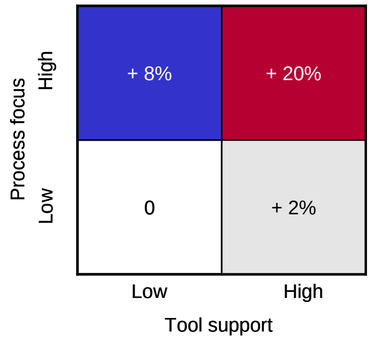

Figure 2: Tools without processes are nothing; processes without tools are not good enough

There are different ALM/PLM environments available today on the market, such as the general purpose solutions from Dassault, Siemens, and PTC which had evolved from mechanical engineering with design and product data management, and IBM and Vector with specific solutions having grown from software centric applications. We will focus in this case study on Vector's PREEvision, a leading automotive PLM solution, due to its use by the company which we portray. Like the other state of the practice PLM/ALM environments, it provides a complete tool suite of integrated data and process management modules [3] (Figure 3).

These state of the practice PLM/ALM solutions provide a repository and editor of different models (Figure 3, left hand side) and ways to link models to reality (Figure 3, right hand side). This allows both a process-oriented way of working (rigid) and a repository-based way of working (flexible, all engineers share a bunch of models of the product). Compared to the more traditional code generation oriented MDE no process consisting of planned model transformations is defined or required.

Based on a rich and extendable data model for features representing the logical and the physical system architecture and the software architecture, PLM/ALM systems provide highly integrated use cases. For instance in configuration and change control, issues are connected to system data objects, the related realization date is fixed in the release planning module, the implementation time and effort are planned in the project management module, the change of the related software parts are managed in the source code management module and finally the test are planned and executed in the test data module.

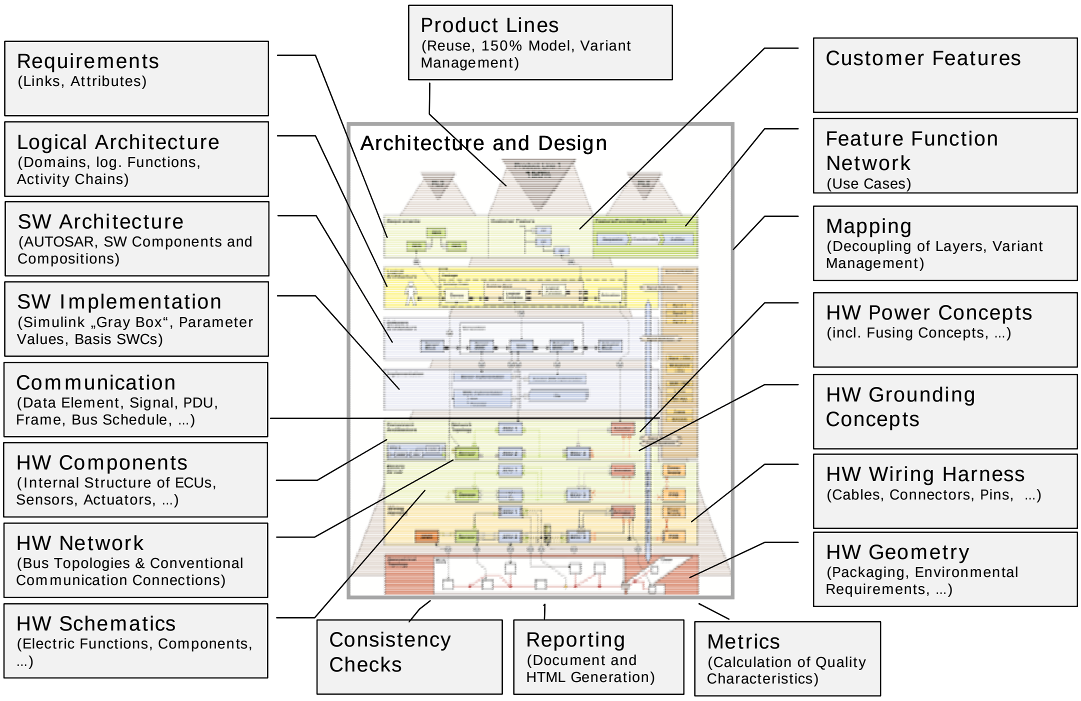

Figure 3: PREEvision solution

Being able to not only reuse information but also guide engineers through complex tasks generates immediate returns by making engineers more flexible and avoiding errors, specifically during last-minute changes and corrections under time pressure. Or, consider the time and effort necessary to move engineers from one project to another. Having standardized PLM/ALM solutions around a standard product life-cycle reduces the learning curve to allow focusing on real technical challenges instead of organization overhead.

3. Industry Case Study

Often the expected benefits from modeling and PLM/ALM tools are often not visible. They are perceived as complex and employees are frustrated and continue working with their current work practices. New processes and interfaces create additional frictions and delays. Our consulting projects show that the root cause is often the same: Implementation of efficient processes with adequate tools support with sustainable results requires profound change management – which is rarely taken into account. To manage such change and to ensure that impacted engineers not only pay lip service but actively support and buy into the new processes and tools, their needs and typical work flows must be understood to avoid that process overheads and heavy tools solutions hinder their creativity. This implies pro-active preparation way before a tool decision is made and good leadership, coaching and support through pilots and roll-out.

We will show a case study from a leading automotive OEM how to successfully introduce PLM/ALM and modeling.

Initial Situation: The main drawbacks of the initial situation were non-connected tool chains and a document driven subsystem and component engineering process with a lot of manually managed interfaces. Therefore the organization had to deal with a time-consuming and expensive reviews/rewriting process, a lot of redundancies and inconsistencies which often were not detected by these reviews, not standardized naming conventions and an ineffective way of developing and acquiring information. Further the information was spread over several data sources and a common version control was not really possible.

Objectives and Solution Approach: The objectives were to increase the engineering efficiency, the quality and consistency of the working products and to shorten the existing lead times in the electronic engineering. The solution approach was to implement a seamless information management of engineering data from function to software and hardware. That is, centralized information (i.e. model) management in a distributed and decentralized engineering process. The main characteristics of such a solution were alignment of processes, single source of information for collaborative teams, reusability of information, using commercially available tools, and continuous development with consistent underlying models.

Change process: To adequately support the introduction and change process we recommend a change methodology that is adapted to specific business goals (Figure 4). PLM/ALM introduction is a long term activity which has to be funded by sufficient competence and budget and competent internal resources. The necessity of a change must be accepted broadly in the organization and the reliable commitment of the top management is a key. The process tool has to fulfill the main requirements of the customer and it must have the potential to grow with the growing needs of the user base. The change partner must be reliable and the chemistry between the main actors should secure the probability to realize a true long term partnership. These are all necessary conditions for success. If one is a false, failure result. But if they are true success is not necessarily guaranteed.

Resistance out of the organization, lack of money, changes in priorities, pressure from the user groups, theoretic discussions – these are not the exceptions, these are normal influences in such a project. The partners have to be able to deal with it and put the best personnel in the position of project leaders and methods/process engineers. Our customer made good experiences with an iterative-incremental development process – short implementation cycles, early validation with a small group of well experienced pilot users and a professional change management guaranteed mature deliveries for productive use.

In the rollout phase normally a small group of convinced pilot users are facing a big group of engineers, which are open for the change in principle, but they are in the conflict to invest on the one hand more upfront time for learning the new environment and to have challenging objectives for their normal work products on the other hand .Furthermore, migration to a new way of working often includes "clean up" of older specifications, which also requires effort. And there is the small group of engineers, who are convinced, that the current, traditional way of working is the optimum and that there is no need for a change at all. We managed this situation with special trainings for key users and initiating “Jump Start Projects”. In these projects the client’s engineers were supported by well known and well with local specific engineering knowledge experienced consultants, which have been trained in PLM/ALM before. Such "Jump Start Projects" are focused and have a predefined duration (typical two months).

Lessons Learned: Utilizing a consistent product life-cycle and process repository is a necessary condition for reducing cycle time, as they reduce frictions of unclear interfaces and responsibilities as well as cutting rework because of inconsistent assumptions and cutting retrieval time for specific documents and work products. In implementing PLM/ALM and modeling support we found the following lessons learned:

- Concept: First improve the process then the tools based on concrete improvement objectives that are set, measured and used to correct deviations. Ensure consistency of features and products with a strong systems engineering. Specifically in distributed collaborative environments we see huge benefits from a single data backbone for consistent requirements, specs and models across all changes

- Development: Evaluate tools under realistic conditions. Agree specific requirements to the process and tools which are then used to drive changes. Support the interfaces to the various components and processes through traceability, automatic consistency checks and test automation.
- Deployment: Manage the changes as they impact the entire organization. Pilot changes, coach and train engineers, highlight power users that will set the pace. Introduce model-based development intelligently and step by step, focus on critical components, continuity of requirements to code and test cases, and improving processes in parallel.
- Operations: Support users and ensure continuous improvement. Measure the implementation, and try in each project ten to twenty percentage points improvement, at the spots where you want to put accents, for example 20% less cost variations, or 10% less cost in the test.

These lessons learned apply to various type of change during the introduction and roll-out of PLM/ALM. They can be generalized for PLM/ALM introduction, or they can be specifically adapted for a micro-level change, such as a change of a modeling methodology.

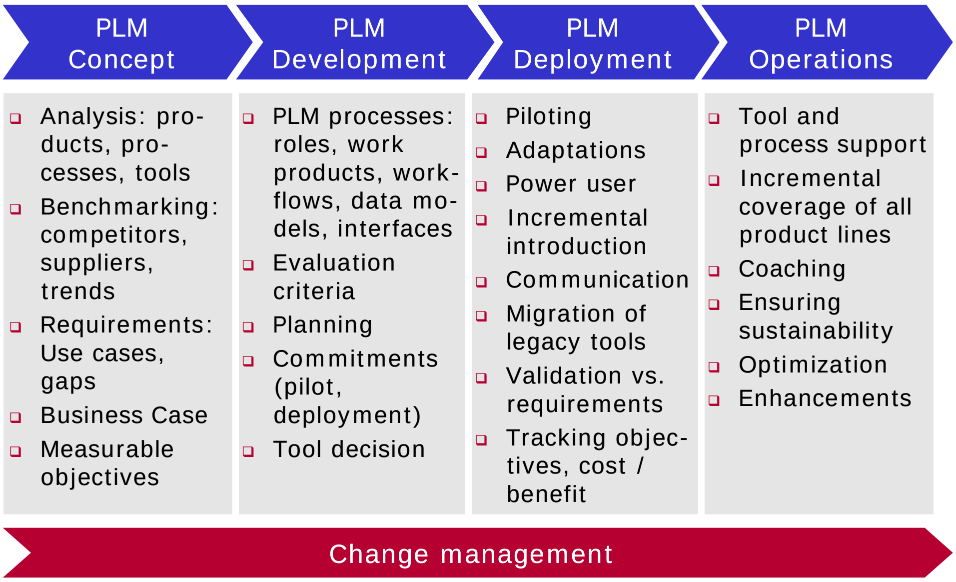

Figure 4: Vector change model for successful PLM introduction

5. Summary and Conclusions

For most companies, there is a wealth of untapped opportunities to cut costs from their development projects. This efficiency levers are sometimes obvious, such as incomplete and wrong information exchange in a distributed product engineering project. At times they are less obvious, for instance if engineering teams work on different variants that emerge only late during system test or product pilots, such as it was reported from a recent multinational airplane project.

Inefficiencies are rampant when engineers are distributed globally and many different tools being used. Concrete efficiency and quality improvements with reduced rework and faster throughput have been showed by applying consistent PLM/ALM and modeling support. The efficiency and effectiveness of engineering processes directly influence engineering cycle time. For instance earlier defect detection in requirements or specs means faster and more comprehensive defect correction.

Without consistent model-based methods and adequate PLM/ALM tools support engineering will be in deep problems within a short time-frame. Increasingly complex requirements must be developed efficiently and with high quality throughout from system engineering to components. Working at a higher level of abstraction and automation will improve productivity and quality. Model-based development will play a crucial role in this evolution. The companies we work with share the same goals: Mastering complexity, improve product quality, shorten development time, and plan functions in different variations and versions for better reuse. The biggest challenges they see are their own learning curves and to keep consistency across features and products. Systems engineering still is undervalued and too often decoupled from application development. Roadmaps are maintained only for subsystems, and thus create variants with an overwhelming complexity. The business case is clear: With a degree of 30% for modeling, error rates and development costs are reduced significantly.

What did we achieve with PLM? Since we consider knowledge management a regular management activity, we followed through like in other improvement projects by looking into performance results from real projects, as well as some process-related aspects, such as knowledge utilization. We found improved quality, reduced cycle time, improved engineering flexibility, reduced overheads improved communication, increasing alignment of processes and tools, and faster ramp-up time and skill management. Initially improved visibility and aligned terminologies and roles already brought huge gains, as they facilitate a borderless solution building inside the company.

What is next in PLM/ALM? Knowledge management must be better linked to business. Aligned business objectives and metrics must guide and monitor the development processes, the product lines and the project teams. Take as an example a mobile phone or game design with lots of embedded software. Being a commodity, business-oriented targets cover return rates or brand loyalty. Defects increase return rate and reduce brand loyalty with devastating business impacts. Looking to projects, products and processes will improve the design away from overly narrow focus on manufacturing aspects towards usability engineering. Knowledge and experience from past projects will be embedded into the underlying design processes. We stress the need for adequate knowledge management as a basis for success in product and solution development – an aspect going well beyond most PLM/ALM approaches of today.

We should not expect that all product and process knowledge and modeling methodology can be codified and made available via PLM/ALM. Personal contact will always be necessary to provide context and analysis. The support system should therefore be extended to facilitate interpersonal communication and evolve towards a global who-is-who not only at the operational product/project management level but also at the tactical and strategic level.

Correctness and completeness of the information is another aspect that needs to be worked on. We are convinced this cannot be achieved by imposing a reporting discipline only. It can only truly be achieved by ensuring that the provider of information is directly benefiting from making the information available. This can be achieved to continuing to evolve the system to ensure information can be captured as early as possible when it is required at the lowest operational level, e.g. by supporting period project reporting in presentation format to avoid information is presented first a set of slides for the project review before it is entered in the system.

Embarking on a state of the practice PLM/ALM solution combined with strong change management triggered by external support had helped to sustainably achieve the anticipated efficiency effects in the different engineering processes across the product life-cycle.

Literature

[1] Ebert, C. and J. De Man: Effectively Utilizing Project, Product and Process Knowledge. Information and Software Technology (IST), ISSN: 09505849, Vol 50, No. 6, pp. 579-594, 2008.

[2] Ebert, C. and R. Dumke: Software Measurement. Springer, Berlin, Heidelberg, New York, 2007, revised edition 2016

[3] PREEvision. http://www.vector.com/PREEvision

[4] Chrissis, M.B., M.Konrad and S.Shrum: CMMI. Guidelines for Process Integration and Product Improvement, ed. 3. Addison-Wesley, Reading, USA, 2011.

Author

Dr. Christof Ebert is managing director at Vector Consulting Services. He supports clients around the world to sustainably improve product strategy and product development and to manage organizational changes. Prior to that, he held global management positions for ten years at Alcatel-Lucent. Dr. Ebert serves on a number of advisory and industry bodies, teaches at the University of Stuttgart and has authored several books including his most recent book "Global Software and IT" published by Wiley.

Contact him at christof.ebert@vector.com

Contact:

Dr. Christof Ebert,

Vector Consulting Services

Ingersheimer Strasse 24, D-70499 Stuttgart, Germany.

Ph.: +49-711-80670-1525

Fax: +49-711-80670-444

Company: www.vector.com/consulting