INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

A METHOD FOR IDENTIFYING IMPROVEMENT POTENTIALS WITHIN PRODUCT DEVELOPMENT PROCESSES

Bruno Gries[1] and Kilian Gericke[2]
(1) Capgemini Consulting (2) TU Berlin

ABSTRACT
When a company tries to improve its product development process one of the first challenges is to become clear about two things: what "improvement" means and where within the overall process it is most necessary. The method proposed in this paper aims at identifying activities within a development process which have a high potential for improvement. It is based on the concept of process efficiency, i.e. the ratio of effort spent vs. value added. While the effort of a development activity can be determined quite easily, the value it adds is more difficult to quantify. Therefore an algorithm is described which uses a Design Structure Matrix (DSM) to calculate the value of each development activity based on the knowledge gained through it and the degree to which other activities depend on it. The method has been successfully applied to a real-life product development process and has received initial positive feedback from industry.

Keywords: Lean Product Development, Design Structure Matrix, process improvement, added value

1. INTRODUCTION
From a company's perspective, the aim of product development is to create competitive products within given cost and time limits [1]. As a business process, it is one of the most critical ones: when performed poorly, serious budget overruns, delayed market entry and flawed products may be the result.
A central aspect of any business process is efficiency, i.e. the ratio of effort spent vs. value added. However, the value added by a development process – let alone an individual development activity – is difficult to quantify. In that respect, product development differs very much from processes like e.g. production where established methods exist to measure and improve efficiency. The "lean" management philosophy, for instance, aims at increasing the efficiency of business processes by avoiding non-value adding activities whenever possible [2], [3]. Still, the concept of "lean" product development has not been widely recognized yet.
In this paper a method is proposed which aims at identifying those activities within a product development process which have a high potential for improvement by comparing the relative effort they require with the relative "value" they create. While not necessarily "lean", this method is intended to facilitate the transformation of a development organization by providing a quantitative yet transparent means of identifying and prioritizing areas of improvement.

2. STATE OF THE ART

2.1 Value stream mapping (VSM)
Both Lean Production [2] and the Toyota Production System [3] are approaches which aim at a reduction of lead-time on shop floor level. A basic concept of lean production is the distinction between value adding, non-value adding but necessary and non-value adding activities. Non-value adding activities are called wastes. Lead-time reduction is achieved through identification and elimination of these wastes. Usually seven different wastes are distinguished [3], [4]:
1. overproduction
2. waiting
3. transport

4. inappropriate processing
5. unnecessary inventory
6. unnecessary motion
7. defects

The identification of wastes is achieved by analyzing and the mapping the value stream in a production system. Based on an understanding of the current state, wastes can be reduced and an improved future state can be defined.

Hines and Rich give an overview of tools which are applicable for value stream mapping. They propose an approach aiming to assist the selection of the right method in the specific context [4].

2.2 Design Structure Matrix (DSM)

A Design Structure Matrix is a squared, i.e. $n \times n$ dependency matrix displaying the vertices of a directed activity graph *G(V, E)*. The method has been first proposed by Steward [5]. Relations (i.e. information flows or dependencies) between activities are displayed in the matrix as shown in figure 1. Sequential relations can be found in the lower left part of the matrix. Relations on the upper right part of the diagonal matrix may indicate iterations in the process [6].

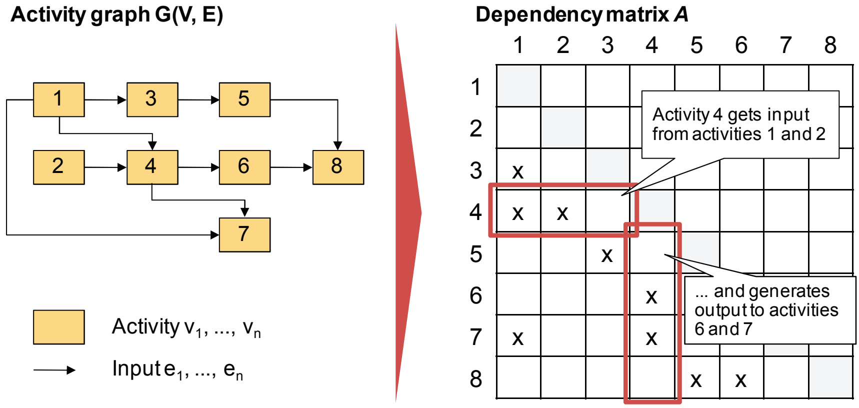


Figure 1. Example of mapping an activity graph in a dependency matrix

DSMs are used for several purposes beyond modeling processes. The method is embedded in numerous approaches used for analyzing processes from different perspectives, focusing on aspects like e.g.:

- Modeling of changes and forecasting the impact, with the goal of a process optimization [7]
- Optimization of the process by reducing the number of iterations and maximization of parallelism of activities [8]
- Analysis of interactions [9]

Thus, a DSM can be used for process modeling, analysis and optimization. The above approaches improve the process by a structural optimization. However, without consideration of the details of each activity or what purpose it serves, the utility of such approaches is probably limited.

2.3 Existing approaches to "lean" product development

Lean thinking in product development

Several authors discuss the application of lean thinking on product development processes. McManus and Millard [10] propose to analyze and map the information flow in product development in analogy to the material flow which is analyzed in a production system (see 2.1). They argue that the value during a product development process increases by creating knowledge. This is in line with the arguments of Browning et al. [11], [12]. Based on this adaptation of lean thinking (or more precisely of the concept of the value stream) to product development the different types of wastes are discussed (see Table 1) [10].

A current study in the German manufacturing industry revealed that a third of the 143 interviewed companies have begun to identify wastes in their product development [13].
The LAI (Lean Aerospace Initiative) group at the MIT has conducted extensive research on the application of value stream mapping in the area of product development. They propose the use of a DSM as a central tool for mapping the value stream through the development process [10], [14], [15].

Table 1. Product Development information wastes [10]

	Waste	Description
1	Overproduction	too much detail, unnecessary information, redundant development, over-dissemination, pushing rather than pulling data
2	Transportation	information incompatibility, communication failure, multiple sources, security issues
3	Waiting	Information created too early or unavailable, late delivery, suspect quality
4	Processing	unnecessary serial effort, too many iterations, unnecessary data conversions, excessive verification, unclear criteria
5	Inventory	too much information, poor configuration management, complicated retrieval
6	Unnecessary Movement	required manual intervention, lack of direct access, information pushed to wrong sources, reformatting
7	Defective Product	lacking quality, conversion errors, and incomplete, ambiguous and inaccurate information, lacking required test/verification

Value of product development activities

A central issue is the definition of the value created during product development [14]. Browning et al. [12] describe the process of value creation in product development as follows: “Product development is a problem-solving and knowledge-accumulation process. Progress is made and value is added by creating useful information that reduces uncertainty and/or ambiguity.” They discuss the inverted relation between the levels of the inherent ambiguity and the created value in product development processes.
Chase [14] discusses the partially philosophical literature about value in product development and concludes that these are not applicable for mapping the value in a real product development process. He proposes a metric consisting of eight types of attributes. In total, he uses 28 attributes to assess the value of a single activity. A problem of such a detailed value analysis is not the analysis of the data; the problem is the survey of the data. While the impact of each individual attribute is comprehensible, the blending of 28 attributes bears the risk of leverage effects, i.e. ending up with all activities having a similar value.

DSM as a method for mapping the value stream in product development processes

McManus and Millard propose a multilevel approach for the mapping and analysis of product development processes (VSM/A). A main goal of their work is to reduce the cycle time of the process. Their approach takes the following steps [10], [15]:

1. Assemble and train VSM/A team
2. Select Value Stream to improve
3. Define Value Stream elements
4. Analyze and map the Current State
    a. Analyze and map the Future State
    b. Analyze and map the Ideal State
5. Implement the new process
6. Continuous improvement

The most difficult step is not the mapping itself (which is done by using a DSM), it is the survey of the current state of the process. This is done in three steps [10], [15]:

7. Map activities and their in-/outputs
8. Capture metrics and characteristics of each activity
9. Consider activity value

They mention the use of the metric proposed by Chase to assess the value but do not claim its use. As shown in more detail, McManus [6] aims to identify the different types of waste directly. Thus, the mapping of the current state e.g. using a DSM and the understanding of the value of each activity is a prerequisite.

2.4 Discussion

Applying a value stream analysis to product development processes enables the user to identify wastes in a process, even in product development processes. The usefulness of DSM as a method for modeling and analyzing different aspects of product development processes has been proven.
As proposed by McManus et al. the combination of both methods is possible. However, some issues still remain. One important issue is the separated assessment of the value for each activity. By doing this, the propagation of information and thus the accumulation of the value are not sufficiently mapped.
The goal of the method presented in this paper is to identify activities with a poor effort-value ratio. This ratio is affected by wastes but not exclusively determined by them. Inappropriate methods and propagation of the created value can affect the ratio as well.
A precondition for analyzing the cost-value ratio is a quantified assessment of the process elements. As argued before such a metric must be traceable and less labor-intensive.

3. DESCRIPTION OF THE APPROACH

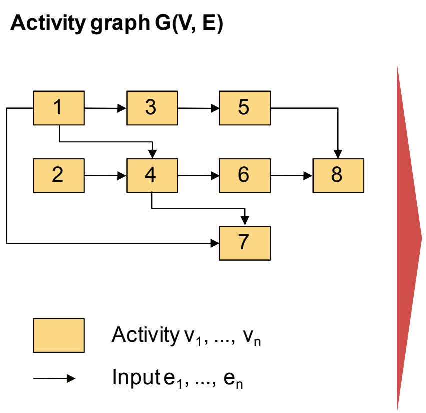
3.1 Step 1: Deployment of the enhanced dependency matrix

The deployment of the enhanced dependency matrix proposed here follows the same basic principles as described in section 2.2. In addition, the enhanced matrix A contains for an activity graph $G(V, E)$ two additional parameters: the expected knowledge increase per activity $k(v)$ and the activity dependency $d(e)$:

$$k(v_i) \Leftrightarrow a_{ii} \quad (1)$$

$$d(\{v_i, v_j\}) \Leftrightarrow a_{ij} \quad (2)$$

According to (1) the expected knowledge increase per activity $k(v) \in \mathbb{R}^+$ is contained in the main diagonal of the dependency matrix A (see figure 2). The values indicate to what extent e.g. a design review, a simulation or the preparation of a paper drawing contributes to the overall knowledge necessary to build the product. It has proven sufficient to define $k(v) \in \{0, 1, 2\}$, where a value of 0 could represent e.g. a purely administrative task, whereas value of 2 would be assigned to a prototype test for example.



Dependency matrix *A*

	1	2	3	4	5	6	7	8
1	0							
2		2						
3	0.5		2					
4	0.5	0.5		0				
5			0.5		0			
6				0.1		1		
7	0.2			0.1			2	
8					0.1	0.1		2

Figure 2.Activity graph and enhanced dependency matrix

Unlike in a classic DSM, where relationships between activities are binary, (2) states that the dependency matrix contains the dependency degrees $d(e) \in \{0, \ldots, 1\}$ of G. In figure 2, the value of 0.2 in column 1 and row 7 of the dependency matrix is equivalent to $d(\{7, 2\}) = 0.2$. Likewise, $a_{7,4}$ holds $d(\{7, 4\}) = 0.1$.

Thus, the values a_{ij} of A indicate to which extent activity v_i is based on results created by activity v_j. In the example shown in figure 2, 20% of activity 7 is based on the outcome of activity 1, 10% on activity 4. The remaining 70% may be interpreted as independent 'gain'. Since the independent gain cannot be negative, the following condition must be true for each $i = 1, \ldots, n$:

$$\sum_{k=1;\,k\neq i}^{n} a_{ik} \leq 1 \tag{3}$$

Thus, all line totals left must be smaller or equal to 1.

3.2 Step 2: Value determination

A major notion of the method proposed in this paper is that the "value" of a design activity e is not only determined by the knowledge increase *k(v)* it yields, but also by the degree to which it is an enabler for subsequent activities. To account for this factor, the following operations are performed for each $i = n, \ldots, 1$:

$$a_{ii} = a_{ii} + \sum_{j=n;\,j\neq i}^{n} a_{ji} \tag{4}$$

$$a_{ij} = (a_{ij} \cdot a_{ii})_{j=1,\ldots,n;\,j\neq i} \tag{5}$$

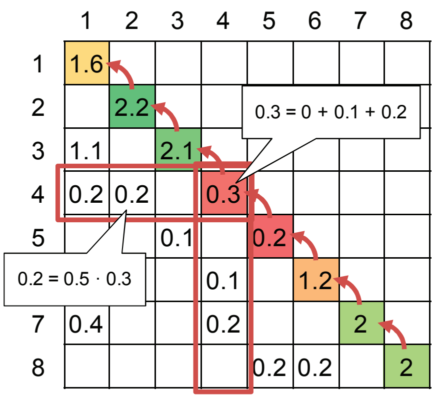


Figure 3. Enhanced dependency matrix after value determination (values rounded)

This iterative algorithm is illustrated in figure 3 (note that for better readability all values are rounded to one decimal). Starting at the lower right corner of the matrix A, (4) does not apply. However, $a_{8,5}$ and $a_{8,6}$ become 0.2 (= 2 · 0.1) according to (5). After repeating (4) and (5) until the upper left corner of the matrix is reached, each element a_{ii} on the main diagonal carries a value that is equivalent to the sum of its knowledge increase $k(v_i)$ plus the total knowledge it contributes to its successor activities. Mathematically, the algorithm is also applicable to matrices with feedback loops (as denoted by values above the main diagonal). However, iteration, i.e. the (multiple, conditional) repetition of whole subgraphs of G cannot be reasonably modeled.

3.3 Step 3: Evaluation

Once the "value" of each activity has been calculated, and the effort (e.g. in hours) has been determined, an Activity Profile can be generated, containing the absolute and relative "value" and effort of each activity (see figure 4). By comparing these two dimensionless parameters of each activity, it is possible to plot a Priority Chart in which the position of each activity indicates its improvement priority.

Activity Profile

Activity	"Value"		Effort	
	abs.	%	h	%
1	1.6	14%	80	16%
2	2.2	19%	40	8%
3	2.1	18%	40	8%
4	0.3	3%	8	2%
5	0.2	2%	8	2%
6	1.2	10%	120	24%
7	2.0	17%	80	16%
8	2.0	17%	120	24%
Total	11.6		496	

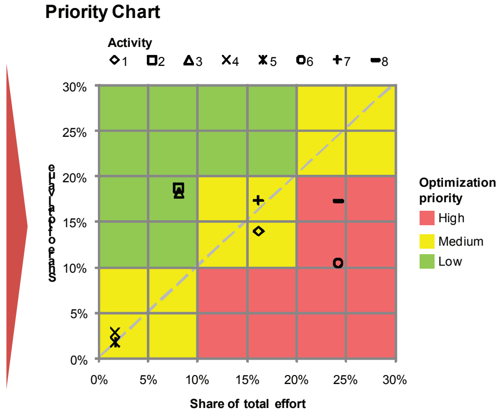


Figure 4.Activities mapped onto a Priority Chart

Obviously, activities which take a high share in the total effort but contribute little to the overall "value" should get the highest attention – e.g. activity 6 which requires 24% if the overall development effort but only adds 10% of the "value".

To improve a "critical" activity, its effort could be reduced and/or its value increased. On the Priority Chart the first option would be equivalent to moving the activity to the left whereas the second option would shift the activity up.

4. CASE EXAMPLE

The approach described in 3. was tested using a real-life product development process with a total of 75 different activities. The process was obtained from the study described in [16] in which a typical 2-year development process for a complex piece of plant equipment had been mapped in two workshops together with key experts from the studied company (see figure 5).

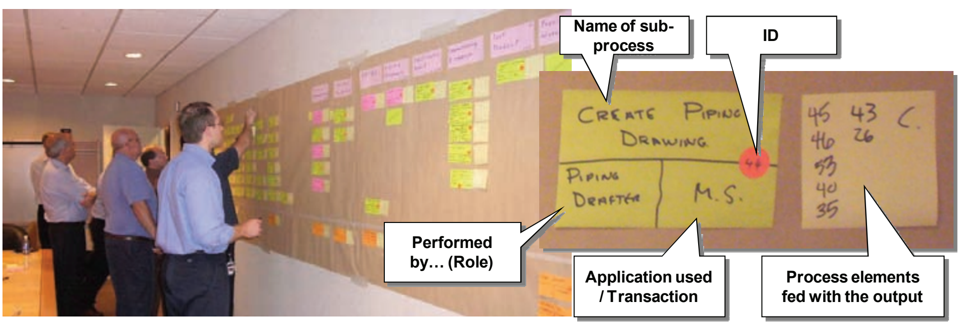
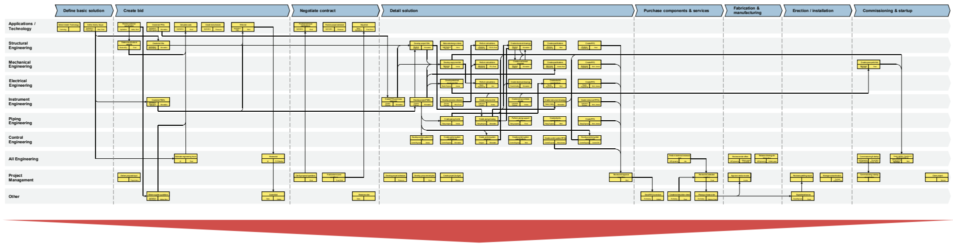


Figure 5. Process mapping workshop

An Excel-based tool has been developed to store the process map in a dependency matrix and to perform all the calculations described above. Note that the values above the main diagonal shown in figure 6 are not to be interpreted as iteration: they result from the structure of the activity graph.

The analysis results were quite convincing. While data cannot be published due to confidentiality restrictions, the outcome of the analysis was plausible insofar as activities that were identified having either a very good or very poor "value"/effort ratio were neither "an obvious guess" nor entirely improbable. Also, early activities did not per se have better ratios than activities occurring late during the process.



	Effort		Value																														
	MDs	%	Abs.	%	Ratio	#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
Select / sketch Technology	3	0.64%	2.00	2.3%	3.61	1	2																										
Define Activity Scope	5	1.06%	3.03	3.5%	3.28	2		3																									
Review customer specification	10	2.13%	0.88	1.0%	0.48	3			0.9																								
Determine structural weights	2	0.43%	2.30	2.6%	6.22	4			0.2	2.3																							
Deliver proposal input	1	0.21%	1.00	1.2%	5.41	5					1																						
Create bid PFDs	10	2.13%	3.37	3.9%	1.83	6						3.4																					
Create bid GAs	10	2.13%	3.43	4.0%	1.86	7		0.7					3.4																				
Create bid P&IDs	10	2.13%	5.52	6.4%	2.99	8		1.1						5.5																			
Obtain supplier quotations	5	1.06%	1.30	1.5%	1.41	9			0.7						1.3																		
Calculate costs	5	1.06%	1.00	1.2%	1.08	10				0.3					0.3	1	0.3																
Estimate engineering hours	2	0.43%	0.30	0.3%	0.81	11		0.2									0.3																
Create bid schedule	3	0.64%	0.00	0.0%	n/a	12												0															
Write bid	5	1.06%	0.00	0.0%	n/a	13							0	0					0														
Review bid	2	0.43%	0.00	0.0%	n/a	14													0	0													
Submit bid	1	0.21%	0.00	0.0%	n/a	15														0	0												
Finalize technical clarifications	3	0.64%	1.00	1.2%	1.80	16																1	1										
Clarify proposal questions	1	0.21%	1.00	1.2%	5.41	17																	1										
Finalize project schedule	2	0.43%	1.00	1.2%	2.71	18																		1									
Finalize terms and conditions	2	0.43%	0.00	0.0%	n/a	19																			0								
Adjust bid	1	0.21%	0.00	0.0%	n/a	20																			0	0							
Receive order	1	0.21%	1.00	1.2%	5.41	21																					1						
Create PFD and mass balances	15	3.19%	2.75	3.2%	0.99	22						1.4																2.7					
Develop project schedule	5	1.06%	1.00	1.2%	1.08	23																							1				
Develop project GAs	10	2.13%	2.87	3.3%	1.55	24							1.4															0.3		2.9	0.3		
Develop project P&IDs	15	3.19%	7.04	8.1%	2.54	25								3.5																	7		
Develop control system I/O list	10	2.13%	2.00	2.3%	1.08	26																									0.4	2	
Develop procurement plan	4	0.85%	0.00	0.0%	n/a	27																											0

Figure 6. Process map and evaluated dependency matrix of a real-life example

In general, conceptual / scope defining activities had the highest ratios, development activities related to the bid and quotation process as well as reviewing activities had the lowest. Activities related to documentation mostly had a ratio larger than one, indicating that this effort is probably well spent.

5. DISCUSSION AND CONCLUSION

Experience shows that when trying to improve the product development processes of a company, designing a new process "from scratch" is almost never an option. Instead, the existing process needs to be analyzed for opportunities where improvements would be most effective. To be successful, the overall approach should involve all key stakeholders in product development, i.e. not only designers and engineers but also experts from production, marketing and sales.

The method presented in this paper can support this approach by identifying those activities within the product development process where relatively little is achieved with relatively high effort. In its application it is fairly practical as only three parameters per development activity are required as input: effort, knowledge gain and dependencies from previous activities.

Even for a rather complex process, these input parameters can be determined in a workshop setting, contributing to a common understanding of the as-is process. The algorithm behind the value determination is sufficiently comprehensible not to be perceived as a "black box". Its quantitative results are easy to interpret and prioritize.

While initial feedback from industry has been positive, the method is subject to two major limitations. Firstly, it is only useful to <u>identify</u> improvement hot spots – how exactly these hot spots can be improved depends. Reducing the effort would be trivial; the complex reality within companies requires a case-based approach to developing effective measures. Secondly, no satisfactory solution has yet been found to apply the method to processes with conditional iteration. For the time being, a possible "work-around" is to reflect iteration by ranges of efforts instead of fixed values. Nevertheless, one important focus of future research will be on refining the method to overcome this restriction.

REFERENCES

[1] Pahl, G., et al., *Engineering Design - A Systematic Approach*. 3 ed. 2007, Berlin: Springer-Verlag.
[2] Womack, J.P. and D.T. Jones, *Lean Thinking : Banish Waste and Create Wealth in Your Corporation*. 2nd ed. 2003, New York: Simon & Shuster.
[3] Ohno, T., *Toyota Production System: Beyond Large-Scale Prodcution*. 1988, Cambridge Massachusetts: Productivity Press.
[4] Hines, P. and N. Rich, *The seven value stream mapping tools.* International Journal of Operations & Production Management, 1997. 17(1): p. 46-64.
[5] Steward, D.V., *The design structure system: a method for managing the design of complex systems.* IEEE Transactions on Engineering Management, 1981. 28(3): p. 71-74.
[6] McManus, H.L., *Product Development Value Stream Mapping (PDVSM) Manual*. 2005, Lean Aerospace Initiative, Center for Technology, Policy and Industrial Development: Cambridge, MA.
[7] Cronemyr, P., A.Ö. Rönnback, and S.D. Eppinger, *A decision support tool for predicting the impact of development process improvements.* Journal of Engineering Design, 2001. 12(3): p. 177-199.
[8] Whitfield, R.I., A.H.B. Duffy, and L.K. Gartzia-Etxabe. *Identifying and evaluating parallel design activities using the design structure matrix*. in *International Conference on Engineering Design ICED05*. 2005. Melbourne.
[9] Eppinger, S.D., *A Planning Method For Integration Of Large-Scale Engineering Systems*, in *International Conference on Engineering Design ICED97*. 1997: Tampere.
[10] McManus, H.L. and R.L. Millard. *Value stream analysis and mapping for product development*. in *23rd ICAS Congress*. 2002. Toronto.
[11] Browning, T.R., et al., *Complex Systems Product Development: Adding Value by Creating Information and Reducing Risk*, in *Tenth Annual International Symposium of INCOSE*. 2000: Minneapolis.
[12] Browning, T.R., et al., *Adding Value in Product Development by Creating Information and Reducing Risk.* IEEE Transactions on Engineering Management, 2002. 49(4): p. 443-458.
[13] Schuh, G., M. Lenders, and S. Hieber. *Lean Innovation: Introducing Value Systems to Product Development*. in *PICMET 2008*. 2008. Cape Town.
[14] Chase, J., *Measuring Value in Product Development*. 2000, Lean Aerospace Initiative, Center for Technology, Policy and Industrial Development, Massachusetts Institute of Technology: Cambridge, MA.
[15] Millard, R.L., *Value Stream Analysis and Mapping for Product Development*, in *Department of Aeronautics and Astronautics*. 2001, Massachusetts Institute of Technology: Massachusetts.
[16] Gries, B., Post-Merger Product Development Integration: A Case Study, in *International Design Conference – Design 2008*, Vol. 2, Dubrovnik, May 2008, pp. 935-942.

Contact: Dr. Bruno Gries
Capgemini Consulting
Kurfuerstendamm 21
D-10719 Berlin
Germany
Tel: +49 (0)162 2343145
Email: bruno.gries@capgemini.com
URL: www.capgemini.de

Bruno is a consultant at Capgemini Consulting. Working for clients from the manufacturing industry, he is specialized in Product Lifecycle Management, focusing on product development processes and tools. He holds a Doctorate in Engineering from TU Berlin.

The authors wish to thank Udo Lange, Jörg Funke and Avinash Gore for their input and feedback.