INTERNATIONAL DESIGN CONFERENCE - DESIGN 2004
Dubrovnik, May 18 - 21, 2004.

DRIVING PRODUCT COMPLEXITY AT AN OPTIMAL DEGREE BY A SET OF KEY FIGURES

I. Graessler

Keywords: complexity management, complexity controlling, key figures

1. Product complexity and goals of measuring

Complexity meets us every day. Impressive examples can be found in nature, such as behaviour of a murder of birds or non linear growth of sea roses. Typical characteristics of complexity are a multitude of interrelated elements, variability and ambiguity. Complex systems are hardly or not at all predictable, barely controlable, viable and thus robust against troubles. As a general rule, they rely on processes connected in parallel (redundancy), each of which is not decisive for the overall function.

In economy of today's developed nations, complexity appears in form of diversity and dynamics. For example, rising product variety must be offered in shortened product life cycles at increasing market turbulences. In order to handle this external complexity, companies react with rising their internal complexity originating from product complexity. Taking the present BMW 7-series as an example, more than 70 electronic control units (ECU's) rely on a software amount beyond 10 Megabytes. Among above 700 vehicle functions around 250000 direct interactions can take place (Saad 2003). Further conclusion upon the correlation between ECU's and complexity is drawn in figure 1 (Weinmann 2002). In comparison to BMW 7-series, the present "Golf V" also already contains 36 ECU's.

These product examples demonstrate the problem we are facing. In order to coordinate and handle product complexity, first of all its extent must be made transparent. Only if reasons for complexity are understood, an optimal degree of product complexity can be found. In order to balance product complexity at its adequate level, representative complexity indicators and self-controlling mechanisms must be found. Put as key figures, they form a complexity measurement system which is boundary condition of successful complexity management.

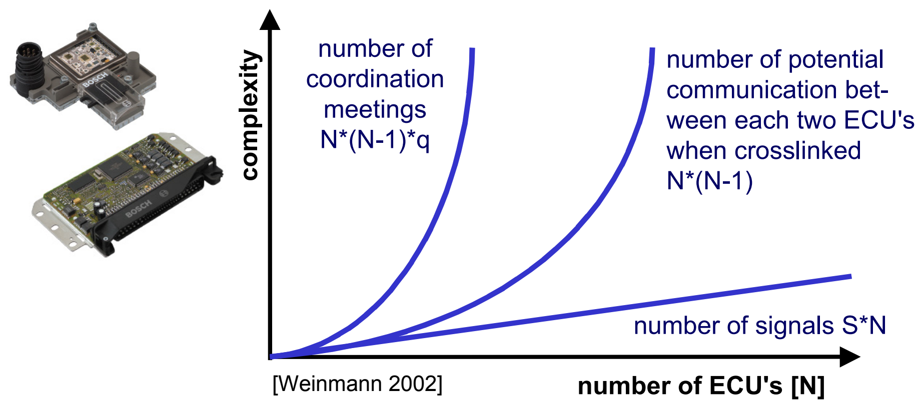


Figure 1. Dimensions of complexity

2. Principles of effective complexity controlling

Effective complexity controlling relies on well defined key figures (chapters 3, 4) as well as successful implementation of measuring system into development processes (chapter 5). Due to nature of complexity, i.e. unpredictability and failing countability, no complete measurement is achievable. Instead, a set of key figures shall outline the decisive aspects of product complexity. Thus self-controlling mechanisms can be put into action by managers and employees.
According to the position argued in this paper, the meaning of complexity controlling not only comprises cost controlling, but also technical controlling. Thus an early warning system is provided. Existing approaches of activity-based costing (e.g. Schuh 1997) suffer from the time gap between generating complexity and detecting its resulting cost. Current approaches of target costing (e.g. Schuh, Schwenk 2001) are limited by forecast uncertainties. Therefore cost-based approaches are only meaningful if they are used in addition to a technical controlling, as such described in the following.

3. Case study: deriving a measurement system

Using an automotive supplier of electronic and mechatronic components as an example, a pragmatic solution approach of complexity controlling in product development is presented subsequently. As main success factor, concerned developers had to be transformed into participating developers who accept the new measurement system. To achieve this, developers were convinced of measurement advantages and involved into each step of deriving the measurement system.
Instead of applying universal solution approaches, a situation specific complexity controlling was worked out. Analysis of present situation revealed a lack of re-use in product development as main reason for over-complexity (figure 2). For instance, insufficient re-use became evident by missing re-usable standards or deficient separation between standard and variable components. Against this background, involved developers agreed upon focusing on re-use as main influence factor. A hierarchy of re-use based on a techniques catalogue was defined as target state.

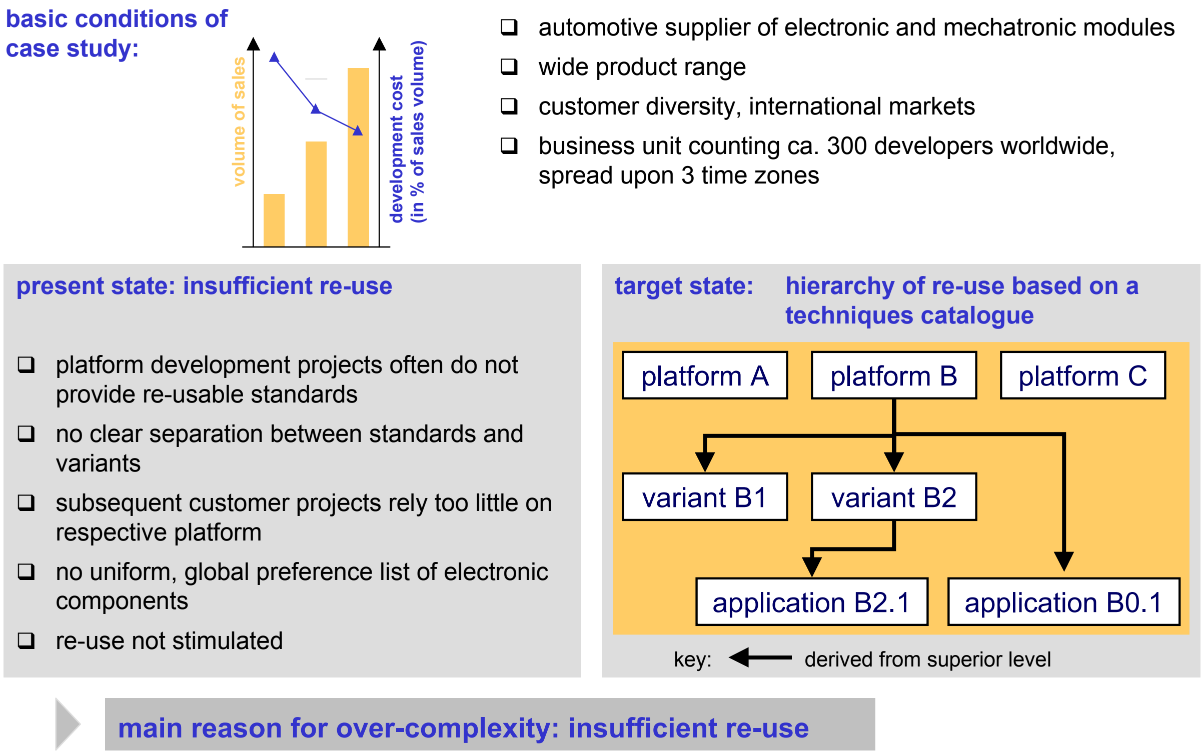


Figure 2. Situation analysis

4. Set of key figures

The degree of re-use is characterized and evaluated by planned re-use and realized re-use. In the course of platform development the amount of planned re-use is fixed. During variant and application projects (customer projects), the planned re-use is realized. Consequently the principle of complexity controlling is based on continuously balancing planned and realized re-use.

4.1 Planned re-use

As boundary condition of measuring planned re-use, data of all existing platforms is systematically collected in a platform list. Each platform identifier is allocated to a business unit, department and product segment. In addition, generation number and platform validity time are given. This overall survey produces transparency and provides input information for subsequent analysis and evaluation by key figures.

For each platform, a product breakdown structure form is created (figure 3). In this form, three different types of re-use are listed. Cross-sectional building blocks (CSBB's) are designed to be used in different product platforms and thus imply the highest level of re-use. Product-segment-specific building blocks (PSBB's) are reusable in variant and application projects, derived from the same product platform. Variable modules (VM's) embody customer- or application-specific variation. The product breakdown structure form is generated in parallel to each platform development project.

In order to describe planned re-use, the following key figures are used on different levels of re-use hierarchy: "re-use of cross-sectional building blocks (K_C)", "re-use of product segment specific building blocks (K_S)" and "share of variable modules (K_V)". K_C is defined as the rate of number of CSBB's and total number of platform modules. In addition to K_C, K_S describes planned re-use within one platform. K_S is defined as number of PSBB's divided by total number of platform modules. Similarly, the share of VM's is recorded by K_V. The three key figures of planned re-use are on the one hand used as milestone criterion of platform concept release. On the other hand, variant and application developers take them as guide and checklist.

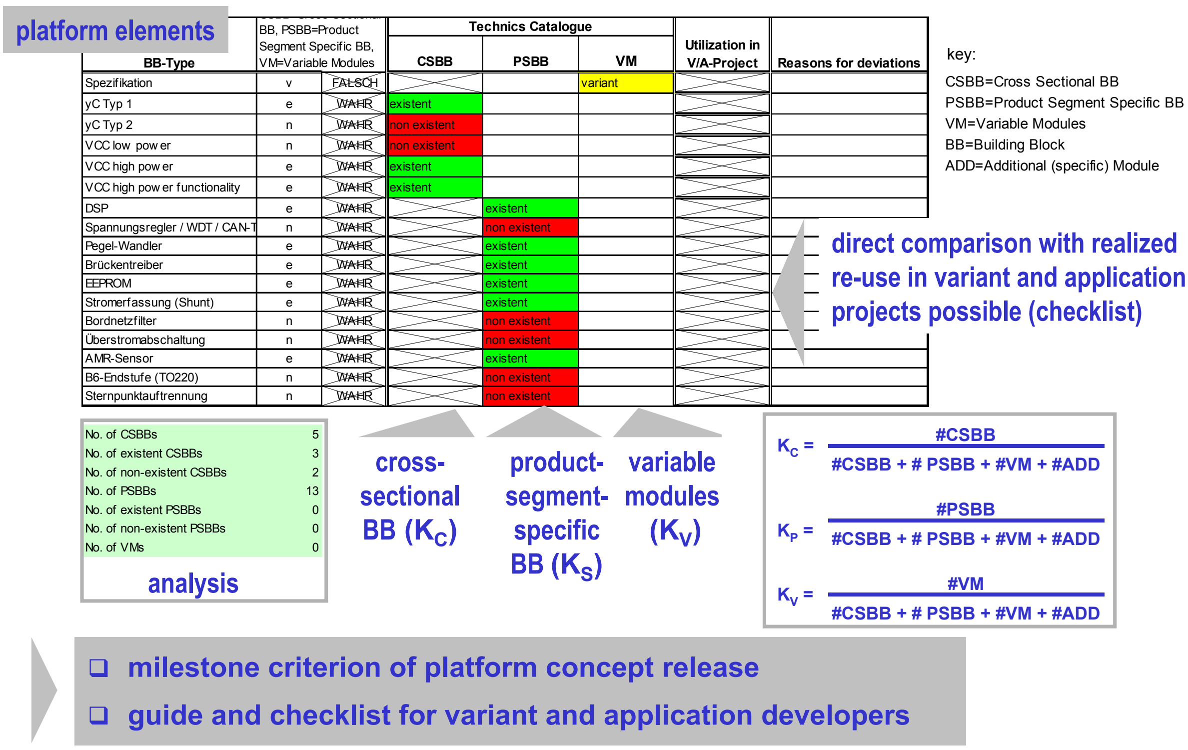


Figure 3. Definition of planned re-use

4.2 Realized re-use

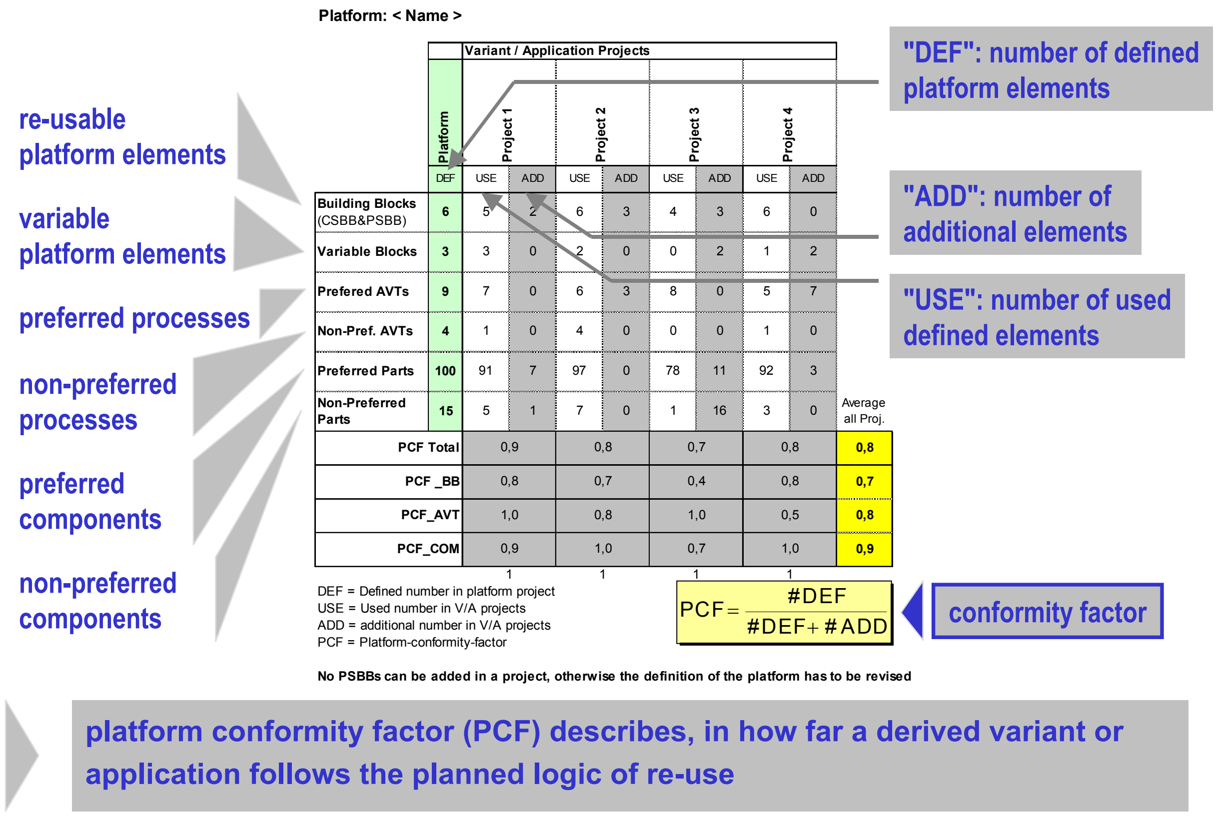
Realized re-use is characterized by the two key figures "amount of derived projects (K_D)" and "platform conformity factor (PCF)". While K_D only permits a general "yes" or "no" statement on level of complete projects, PCF analyzes in detail, how close realized degree of re-use meets planned re-use on module level. K_D is defined as the percentage of variant and application projects which are derived from product platforms. In order to generate K_D, all variant and application projects are listed and assigned to the respective platform project. Thus, customer projects which cannot be assigned are directly detected. Counting the numbers of derived and not derived customer projets, key figure K_D is build up (figure 4).

amount of derived projects

$$K_D = \frac{\text{\#derived customer projects}}{\text{\#customer projects}}$$

Figure 4. Definition of amount of derived projects (K_D)

Addingly PCF describes, in how far a derived variant or application follows the planned logic of re-use. As prerequisite, product structure of customer project is compared with product structure of respective platform using 5th column of product breakdown structure (figure 3). For each customer project, the number of used (USE) CSBB's, PSBB's, VM's, preferred processes and electronic components is counted and compared with the number defined (DEF) in the respective platform. Also, the number of additional elements (ADD) which have been supplemented specifically for a customer project is taken up (columns "DEF", "USE" and "ADD" of each customer project in figure 5). Using this input data, PCF describes share of number of defined elements in respect to total number of elements. The term "elements" implies that not only product modules are reviewed, but also production processes and purchased parts (electronic components). Regularly reporting the presented set of key figures, planned re-use is steadily compared with actual re-use.



	Platform	Variant / Application Projects								
		Project 1		Project 2		Project 3		Project 4		
	DEF	USE	ADD	USE	ADD	USE	ADD	USE	ADD	
Building Blocks (CSBB&PSBB)	6	5	2	6	3	4	3	6	0	
Variable Blocks	3	3	0	2	0	0	2	1	2	
Prefered AVTs	9	7	0	6	3	8	0	5	7	
Non-Pref. AVTs	4	1	0	4	0	0	0	1	0	
Preferred Parts	100	91	7	97	0	78	11	92	3	
Non-Preferred Parts	15	5	1	7	0	1	16	3	0	Average all Proj.
PCF Total		0,9		0,8		0,7		0,8		0,8
PCF _BB		0,8		0,7		0,4		0,8		0,7
PCF_AVT		1,0		0,8		1,0		0,5		0,8
PCF_COM		0,9		1,0		0,7		1,0		0,9
		1		1		1		1		

$$PCF = \frac{\#DEF}{\#DEF + \#ADD}$$

Figure 5. Definition of platform conformity factor (PCF)

5. Integration into development processes

In order to install the proposed measurement system in line organization, a concept of integrating measurement system into business processes is neccessary. Such an integration concept covers responsibilities, reporting and decision structures. In figure 6, the resulting concept is illustrated. During conceptual design of customer neutral platform development the appropriate degree of re-use is planned. As all platform standards, i.e. CSBB's, PSBB's, VM's, preferred production processes and preferred components, are stored in a universal techniques catalogue, new proposals for standards are identified. Proposals must be released by a technical committee before they become a part of the techniques catalogue. If open questions are left, proposals are reworked. As pointed out in chapter 4.1, key figures of planned re-use are used as milestone criterion of platform concept release.
Once the platform concept is released, derived variant or application projects can use the planned re-use as development guide. First of all, specific product requirements are compared with planned re-use. If deviations from techniques catalogue are detected, they have to be released by the technical committee. As next step, the planned conformity factor of the customer project resulting from comparison has to be released at the milestone "conceptual design". From this point on, realized conformity factor is continuously reviewed against planned conformity factor.

6. Conclusion

"You cannot control what you cannot measure" (DeMarco 1982)

Nature of product complexity implies that it may never be fully described or counted. However, the proposed set of key figures offers a pragmatic solution approach for daily development. Re-use is enhanced by creating transparency and comparing planned with realized degree of re-use. Thus product complexity is pro-actively driven to an optimal degree and high cost reduction potentials are opened up.

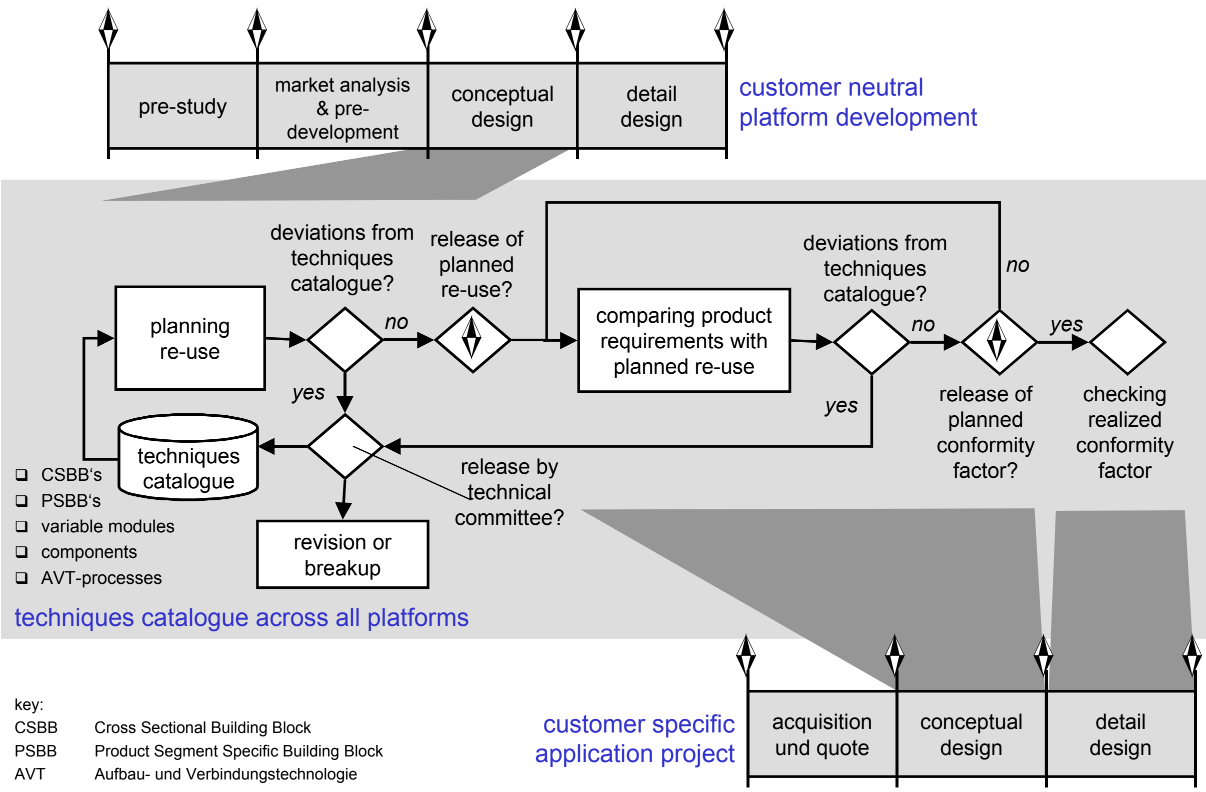


Figure 6. Organizational embedding

The following three factors ensured success in the illustrated case study:

1. gaining acceptance of concerned developpers and making them participate in developping and testing the measurement system,
2. coherent measurement system which can be applied without noteworthy supplementary efforts,
3. mandatory integration of measurement system into line organization.

However, even if these success factors are fulfilled, one has to keep in mind that measuring is a neccessary, but not a sufficient condition of successful complexity management.

References

Weinmann, U., "Anforderungen und Chancen automobilgerechter Software-Entwicklung", 3. Euroforum Fachkonferenz Software im Automobil, Stuttgart, 01.07.2002

Saad, A., "Software im Automobil, Ausgangslage, Zielsetzung und Aufgabe der BMW Car IT", Automotive Electronics, 1/2003, p. 22-26

Schuh, G., "Wohin bewegt sich das Kostenmanagement?, Methoden zur verursachungsgerechten Zuweisung von Gemeinkosten auf die einzelnen Produkte", krp, 41. Jg. 1997, H.1, p. 34-39

Schuh, G., Schwenk, U.; "Produktkomplexität managen, Strategien, Methoden, Tools", Hanser, München, Wien, 2001

Privatdozentin Dr.-Ing. Iris Graessler
Robert Bosch GmbH
Director CIP Coordination
P.O.Box 1342
72703 Reutlingen, Germany
Telephone: ++49/7121-35-1543, Telefax: ++49/7121-35-4707
E-mail: iris.graessler@de.bosch.com