ROLE OF THE INFORMATION AND KNOWLEDGE IN DIGITAL ENTERPRISE TECHNOLOGY

M. Bossmann, H. Bley, N. Avgoustinov
Institute of Production Engineering / CAM, Saarland University, Saarbrücken

The globalization and its consequences have driven the competition to a new (probably again temporary) peak: the Internet has made not only the (main) information about every product, but also its comparison with its competitors just a few clicks away. Since the customers' demands for new functionality, high quality, fast delivery, low prices and good service are permanently increasing, for remaining competitive the enterprises have to invest in novel technologies, allowing high quality despite low production costs and good management of their own know-how. Nowadays the know-how and the knowledge around the product development process represent the most important aspects of companies to stay in market. It is essential to speed up product development process with a high quality to get competitive advantages. At present, several methods exist to support the Simultaneous Engineering (SE). The speed and the flexibility of the development process depend on the information, its representation and its transfer. The goal of our new approach is to minimize the waste of time by improving the support for the data transfer of all necessary information between the departments, involved in the development.

1. INTRODUCTION

The goal of our new approach is to minimize the waste of time by improving the support for the data transfer of all necessary information between the departments, involved in the development. So it is possible to automate different activities of the departments by using a standardised information format. A next step towards a better parallelisation is the definition of milestones along the product development process. This concept allows focusing the work on the data exchange at certain stages for reaching a higher stage of maturity. Main advantage of our approach is a shorter development time that is achieved by transparent information management. In doing so, it is possible to produce initial or enhanced information for different tasks in product development. Then, the result is a better integration of all involved departments that gives us also the opportunity to create new knowledge, based on the information flow and on the discussions among the department experts.

2. DEFINITIONS

On the lowest level of all kinds of information and knowledge structures stays the data. Each of the levels above can be viewed as a data derivative. Since these derivatives play an enormous role in the modelling, we shall explain briefly some of them.

Under *data* we shall understand *strings of (ordered) symbols*. These symbols are represented in computers by numbers, and in turn, the numbers are represented by binary digits or bits. Apart from bits, the numbers are the smallest "building block" in the representation of data and data derivatives – including algorithms – in a computer. An example of such a string of symbols is "3.1415".

The connection or relation between groups or pieces of data shall be called *meta-data*. Given the strings of data "5.003" and "5.02", we can connect them into a relation, for instance by the sign "smaller than": 5.003 < 5.02. Thus, the "<" symbol has special meaning and is meta-data in this case.

The *information* emerges from interpreting data. The interpreting means that each piece of data and meta-data is connected or put into a relation with already known facts as well as with all other pieces of data. Thus, the example above would be interpreted as putting two numbers in a relation, saying that the second one is greater than the first one. For achieving this, the interpreter (human or machine) should have some *a priori knowledge* – i.e. to be able to read the numbers and to understand the meaning of the sign "<". This knowledge is often called *context* or *background knowledge*.

Since the context always plays a crucial role by gaining the information from given data, another possible definition for information is *data in certain context*.
Another possibility to define the term information is as a combination of meta-data and (groups of) data that are to be connected/related, for instance:

$\pi \approx 3.1415$

Such combination of data and meta-data is usually named an *attribute-value pair*. When representing more complex information it is possible to nest attribute-value pairs by using a given pair as the value of another pair.

When the value of an attribute-value pair contains just data (i.e., there is no nesting), such pair can be named *basic* or *substantial* attribute-value pair. Independently of their representation, basic attribute-value pairs can be viewed as the smallest units of information.

The ability to gain new information from already existing information or data shall be called *knowledge*. For instance, if one knows that the circumference of a circle is always $2\pi R$, he is able to deduce that when the radius R=1, then the circumference will be 2π.

The intelligence is a quality that is inherent mainly to human beings. When we say: "this program is intelligent", we mean that in a given situation it behaves or attempts to behave similarly to a human being, which is in the same situation. We shall view a software program as being intelligent if it possesses at least one of the following:

a) Ability to complete maximum tasks with a minimum instructions from the controlling user or program;
b) Ability to guess what action is desired/needed in any given moment and either propose or perform it;
c) Ability to request or find alone any missing or invalid data or information.

It is difficult to think of a way to measure or compute the intelligence of a program that would allow an objective comparison with other programs.

The software models are built-up from data, data-derivatives and (possibly) code. Therefore, many of the model traits depend on the traits of the underlying data and its derivatives, as well as on the chosen representation. For that reason, it should be kept in mind that most of the properties of software models are represented through real numbers, and when these numbers are approximated in their computer representation, the respective models could be badly influenced. Software models can use additional data for specialization (concretization) and communication. Software models can use bound or built-in code (as a special kind of data) for implementing intelligence.

3. WHERE THE INFORMATION COMES FROM

The development of every product starts with an idea, which can in extreme case be described by one word only – for instance "car". The amount of data, necessary for representing this word is modest – in our example only 3 bytes – but the information, related to the idea and its related notions could be enormous.

As the idea evolves, though, the information describing and supporting it increases continuously – first the aim is stated, then the requirements are determined, then the functionality is specified, details are described and so on. The information increases – often even exponentially – with each step of the workflow, with each phase of the product life cycle until the product reaches the market and even after that.

4. AIMS OF INFORMATION TRANSFER AND STORAGE

The objective of a fast development process can be achieved only by an improved information exchange (Alt, 2002; Mäntylä, 1996;Thoben, 1997). For this, milestones by which the relevant information has to be transferred must be defined. Modern CAx Software systems that are used along the product development process make it possible to speed up the special tasks of the different departments (VDI 4499, 2006; Wiendahl, 2002;Schiller, 2002;Westkämper, 2003; Alberts, 2003; Bracht, 2005). The three-dimensional model of the products as well as the use of parametric modelling approaches has performed a decisive step in CAD technology (Bär, 2001; VDI 2218, 1999; Weber, 1996; Cuiper, 1996). New modelling methods make a faster product design possible. Design in Context has to be named as an example in this area that reduces the placing effort at the assembly of the product, since the product will be modelled into final position and use other components' geometry to create geometry in a simple way, e.g. if the designer must create a bolt in a hole, he has the possibility to use the geometry (a circle) of the borehole without troubling about the diameter. The use of feature elements accelerates the process of modelling since routine activities or standard elements can be finished off by bringing in a feature concept. Databases and libraries of standard components can be prepared by suppliers or the own manufacturing to achieve a faster product design. By the use of database elements a standardization (von Langsdorff, 2003) of the products is carried out and often synergy effects in the context of orders or the storage of the elements arise. These synergy effects represent the positive factor of a price diminution since discounts are negotiated at larger order quantities or the own production can establish bigger batches, through what the fixed costs of the production can be more distributed on a larger number of products. A faster progress in the information gain has the consequence that the data transfer can be started earlier. The information gain has to be put with the maturity degree of the planning section at once, through what the release and the transportation of the product is achieved faster to a higher level (maturity degree). This gain in time can be used by the enterprises to put faster new product variants on the market to increase the sale and therefore the sales volume of the enterprise. This effect is strongly distinctively performed at products with a short life time and a lot of evolution steps. An example forms the mobile communication industry which had to develop new functionalities and incorporate them in the respective devices in very short intervals during the last few years. For being a market leader it is essential to put new innovations on the market as fast as possible, because the second one is the first loser. Enterprises that want to exist in such a turbulent environment are interested in optimizing their product development to commercialize product innovations faster and take market shares away from the competitors. So, (Alt, 2002) it is of decisive importance that the tasks of the departments are coordinated with each other and the information importance is indisputable. If the company does not deal with the information as the highest good, then there are information losses that are equate with losses of market shares.

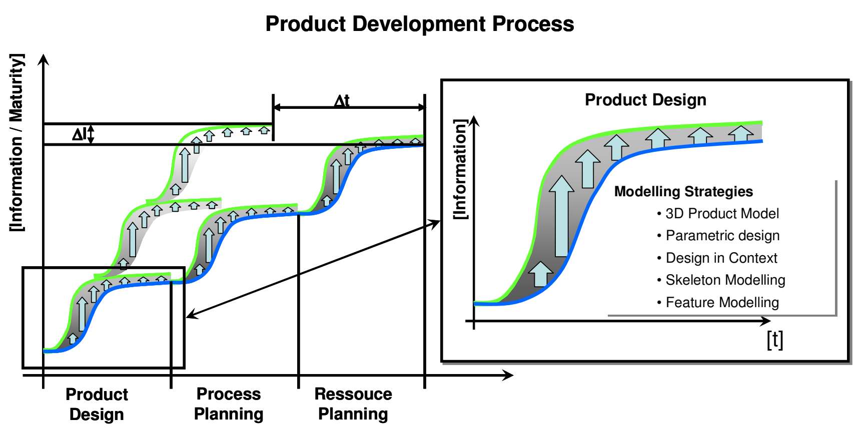

Figure 1 Product Development Process

The time of no information progress (t_0) must be minimised ($t_0 \rightarrow 0$), so that a maximum time saving can be obtained. Achieving $t_0 \rightarrow 0$ is not realistic, because there are insignificant preparations, they can not eliminate such as the data conversion of the interface file to prepare it for further activities and tasks.

The planning activities must be subdivided into different task classes like the SMED (single minute of the die) of the machine rigging. The tasks get subdivided in sub-tasks that represent a close connection to the product and development maturity with the result that these tasks represent the limited factor of the development process. But there are also tasks that can be worked out in rough structures parallel to other tasks. One can obtain time savings by a good structured organization of the planning tasks that constitutes in the best case the minimised waste of time in the product development process. Another advantage by a better preparation for the departments on their tasks is the minimal conversion effort by preparing the input data. Doing so, these method effects no real parallelisation. Therefore the aim is to support planning activities by the product design in a better way. For this reason the tasks of the departments must be subdivided, so that a true parallelization, based on semi finished tasks that represent initial information for other departments, can succeed.

5. STRATEGY FOR PLACEMENT OF INFORMATION AND KNOWLEDGE

All relevant information of a given product is gathered and stored in the so-called product model. In ideal case, the product model is created with the inception of the product and then gradually extended to cover each new need, arisen during the product development.

A product could be so special as to cause the re-development of its production. This process can begin as soon as it is at least partially clear how should the product be manufactured and can give rise to a so-called informational production model. It is derived from the product model and is, therefore, tightly bound to it.

During the product development the product models are usually processed by different CAx-systems. The reason for this is that each of these systems is qualified to solve exclusively tasks, belonging to that phase of the product development cycle for which the system is developed. In (Avgoustinov, 1997) pp. 61-69, the *phase qualification degree* (QD_{Ph}) of a given CAx-system for a given phase (expressing its suitability to serve all specifics for that phase tasks) is defined as the ratio between the cardinalities of the set of system's relevant functions and the set of phase's specific tasks. The later set is represented in Table 1 by the number of columns for the phase, while the former set is represented by all "√" symbols within a corresponding to the respective CAx-system row. Should a CAx-system exist, having a QD=1 for all phases at the same time, it would be called computer integrated manufacturing system (CIM-System). Since due to its exploding complexity no such system is expected to exist in observable time, the processing of each model is performed by several CAx-systems, whereas each system is used for these tasks for which it is best suited.

Despite numerous attempts to establish a common format for representation of the model data or at least for the information exchange, every CAx-system still uses its own format due to historical, strategic, commercial or technical reasons. Therefore, a conversion of the information takes place during (or after) its transfer from one system to the next. When using such conventional workflow for product development, the mentioned transfer and conversion consumes many resources but, at the same time, seems unavoidable.

Table 1: Qualification Degree of Different CAx-systems

	$Phase_{k-1}$					**$Phase_k$**						**$Phase_{k+1}$**					
	$Task_1$	...	$Task_i$	...	$Task_{N(k-1)}$	$Task_1$	$Task_2$	...	$Task_i$	...	$Task_{N(k)}$	$Task_1$	$Task_2$	...	$Task_i$	...	$Task_{N(k+1)}$
CAx_1		...		...		√	√	...	√	...	√			...		...	
CAx_2		...		...	√	√	√	...	√	...	√	√		...		...	
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
CAx_l		...		...		√	√	...	√	...	√	√	√	...	√	...	√
	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
CAx_L		...	√	...	√	√	√	...	√	...		√	√	...		...	

In our view, one of the most prospective approaches for overcoming the problems, arising from the mentioned transfer and conversion, is to re-orient the development process and base it on (model) components.

We see two main advantages of the component-based development: a) they are designed for easy integration with other components, and b) in cases where the integration of two specific components is not foreseen, it seems easier and more efficient to achieve it by communication-based model integration instead of by full model transfer and conversion.

The functionality and properties of any product, which are important for/during most phases of the product lifecycle, deserve special attention. Since they are often used as a core, around which the other characteristics and functionality of the product are formed, they are called *core functionality* and *core properties*, respectively. They are modelled as early in the product development as possible, and are used to build the *skeleton* of the future product or process model. Thus, the exchange and conversion problems of the conventional CAx-system-based development affect at worst the modelling of core properties and functionality. The component-oriented model development would allow to avoid this and to achieve overall improvement.

6. CONCEPT AND IMPLEMENTATION

The design process starts accordingly to the standard with an idea and then it continues with the development of function structures that solve the requirements to the product. In the next step these function structures will be transformed into rough product geometry elements. Numerous function structures give the possibility to start planning activities that get filled progressively based on the development level and additional information about the product. Basic processes can be derived from the function structures (Bley, 2004). On this stage it is already possible to limit the resource choice and to realise first production concepts based on the process plan.

The introduced elements, which standardize the product structure form the general conditions, build the basis for achieving earlier and faster information transfer during the product development. Based on the characteristics of the parts, the manufacturing simulations receive the needed relations and parameters that must be set. Resources, ensuring that given functionality will be produced, must be assigned to the simulations of manufacturing – for instance, component geometry must for the required tolerances.

Afterwards, a tolerance check can be carried out as soon as this parameter is set. In addition, the part production and the assembly planning are also supported by the new elements, since the deposited information represents boundary conditions for the manufacturing and the assembly (Bley, 2004; Bley, 2005; Bley 2005). Thereafter, the contact surface matrix, the contact surface graph and first assembly simulations can be built up automatically, without losses of time. Based on automatism, product changes can be analysed faster. Product models and production models represent flexible structures that can be used for analyses in case of product or production changes. The fear of today's organisations generating an effort that perhaps would not bring any progress avoids a realisation of a real parallel development process.

Today, there are a lot of software tools that support the development, but there are only a few methods for realising an integrated workflow with automations, minimising the respective efforts and the risk of waste of time and money (Ehrlenspiel, 1995). As mentioned in the introduction, the enterprises must remain competitive and therefore they must apportion the costs on their products. One way to achieve this is to diminish avoidable effort for being able to produce economically. Therefore, the aim should be to realise concepts and methods that minimise the explained risk and speed up the development process by launching the mentioned planning activities as early as possible and supporting them by automatism. Firstly, the results of the automatisms must always get validated and their correctness shall be ensured, when necessary – by means of manual intervention. Secondly, the automatism only can be used for the support of routine activities, because the implementation effort justifies itself in this area. With such an approach it is possible to attain the advantage of a faster information gain in the product development, since the time is minimised without information progress and therefore the consequence will be earlier production start. So, it is possible to bring innovative products on the market before the competitors with the advantage of larger profits and an image boost. The resulting time savings can be reinvested in further development of the products, the manufacturing and the assembly, leading thus to a higher quality at a lower costs. So, there are no losses of market-shares resulting from bringing the products on the market late.

Due to the integration of control loops in product design and production planning one can expect that the knowledge increase is slowed, but the information quality of the separate processes is higher, since they are coordinated with each other. This leads to a much better information exchange between the departments. This process is also a subject of a learning curve like all other activities, therefore the effort and the time will be reduced at every usage.

The data interchange at earlier times causes a shift of the activities by the design process that have to be subdivided. The final geometry of the product does not represent the maturity level of the advantage of an earlier data transfer. After the design stages, functional modelling, basic geometry and final geometry a data transfer to other departments is probable. So already the functional geometry definition allows the definition of rough structured process plans and planning tasks. Figure 2 shows the information progress in the individual phases of the product development by the use of simulation tools which have an improved protected stage of development as consequence. Another advantage in the product development is obtained by improved dealing with knowledge and knowledge management, realising an available transparency and storage.

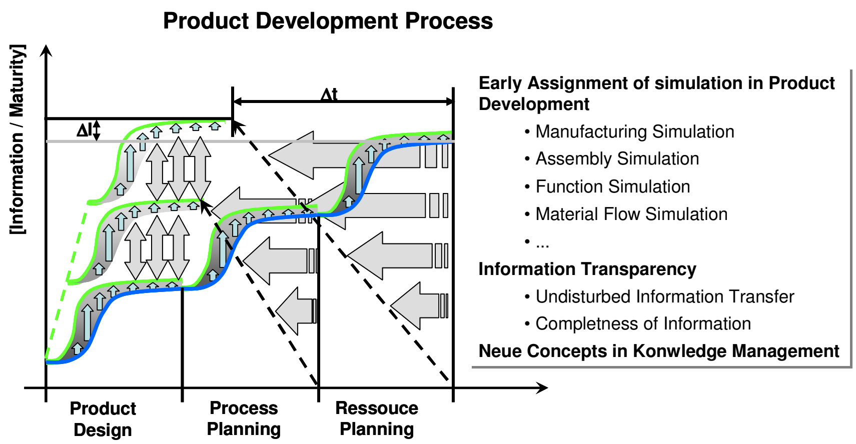

Figure 2. Optimized product development process

Remembering Table 1, such an integrated workflow is based on the data handling by software tools, so there are less barriers by using software tools with a lot of CAx modules that are accessing the same data base and data structure. If the company uses different tools of different software provider, it is important to prepare a good structured transfer file or transfer medium with all needed information to realise the integrated workflow. **Figure 3** shows the skeleton concept that deals with the requirements. The design phase is standardised with new elements called skeleton elements that represent functional geometry areas of the product. These elements represent also a motion control element based on the global coordinates of the product model and its orientation. Automatism supports the design task by interpreting the skeleton elements into basic geometry that can be derived from the functional model. On the one hand, the basic geometry model can be used in assembly planning to find collision free assembly sequences and to support the positioning of the resources.

This result can be achieved by the three layer concept of collision free assembly planning (Bley, 2006). On the other hand, it is possible to realise a connection between the flexible product model and the process simulation of the manufacturing process, realising a flexible product and process model that can be enhanced by resource information to create a Product-Process-Resource-Model (Bley, 2005). Other advantages of the presented concept are possible enhancements that use the feature structure as an integration platform. So e.g. the quality management can use the feature geometry and its semantic information to integrate a defined number of measurement points at certain areas of the feature element, supporting the programming of (coordinate) measuring devices.

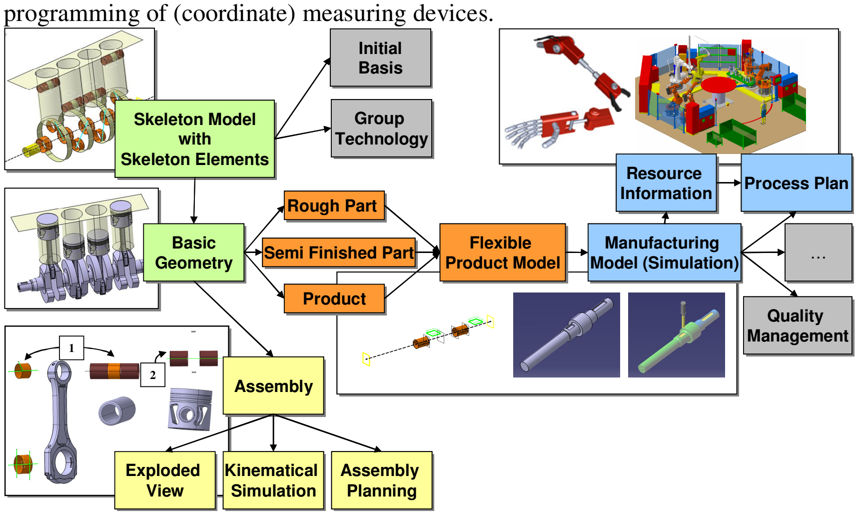

Figure 3. Information Transfer Concept

7. SUMMARY AND FUTURE WORK

Information, information transfer and information storage becomes in the last years a competitive factor of the companies. So it is very important to handle information as one of the highest goods of the company. The presented concept achieves a "real" parallelisation of the different task of the product development process by automating routine work of the different departments. Information transfer is based on the first digital product model of the design department, the functional model. Generating the semantic information by using the feature concept and database knowledge represents a possibility to create initial information for the other departments that can start generating rough process plans and simulation models. So the product model can be analysed in different ways by different departments in an early stage of the product development process. The rough process plans and the simulation models will be more detailed during the development process. In doing so, a fast raise of maturation is forced and the waste of time is minimised by the reduction of the effort of the information transfer, An other important aspect is the consequent use of modern methods in product design, manufacturing and assembly planning and also in resource planning to raise the information level as fast as possible. Future work is the integration of other task along the development process to realise a totally integrated product development environment that deals with the possibility of the integration of control loops between all departments to estimate modification in all aspects.

~

8. REFERENCES

1. Alberts A, Nowicki L. Integration der Simulation in die Produktentwicklung- Neue Möglichkeiten zur Steigerung der Qualität und Effizienz in der Produktentwicklung, Symposium "Simulation in der Produkt- und Prozessentwicklung" 5-7. November 2003 Bremen, pp. 141-147.
2. Alt G, Bill H, Machnig M. Innovation – Technik – Zukunft Die Wissens- und Informationsgesellschaft gestalten, (in German), 2002.
3. Avgoustinov Nikolay. Minimizing the Labour for Exchange of Product Definition Data Among N CAx-Systems. Saarbrücken, Germany, Universität des Saarlandes, 1997.
4. Bär T, Haasis S. Verkürzung der Entwicklungszeiten durch den Einsatz von Skelett-Modellen und Feature-Technologie, VDI-Bericht Nr.1614, Düsseldorf, Germany, 2001.
5. Bley H, Bossmann M. Improved Manufacturing Planning based on Localisation of Product Synergy Effects by the Use of Feature Technology. Proceeding of the 38th CIRP International Seminar on Manufacturing Systems, Forianopolis, Bazil, 2005.
6. Bley H, Bossmann M. Standardisierte Produktmodelle für die automatisierte Montageplanung Featurebasierte Montageplanung unterstützt den Simultaneous Engineering-Prozess, wt Werkstatttechnik - Ausgabe 09-2005, pp. 627-631, 2005.
7. Bley H, Bossmann M. *Automated Assembly Planning Based on Skeleton Modelling Strategy*, Precision Assembly Technologies for Mini and Micro Products, pp.121-131, IFIP International Federation for Information Processing, Springer Verlag, 2006.
8. Bley H, Bossmann M, Zenner C. Flexible Process Models in Manufacturing based on Skeleton Product Models. Preprints of IFAC-MIM Conference on Manufacturing, Modelling and Control, Athens, Greece, 2004.
9. Bley H, Bossmann M, Zenner C. Advances towards an Integrated Product and Production Development Process. Proceedings of 2. German - Israeli Symposium for Design and Manufacture - Advances in Methods and Systems for Products and Processes, Berlin, Germany, 2005, pp. 129-137.
10. Bracht U, Schlange C, Eckert C, Masurat T. Datenmanagement für die Digitale Fabrik, wt Werkstatttechnik online H.4, 2005, pp. 197-204.
11. Cuiper R, Feldmann K, Roßgoderer U. Rechnerunterstütze Parallelisierung von Konstruktion und Montageplanung. ZWF 91 7-8, pp. 338-341, 1996.
12. Ehrlenspiel K. Integrierte Produktentwicklung. München, Hanser, 1995.
13. von Langsdorff P, Rist T. Aufbau innovativer Produktionssysteme durch Standards und Plattformkonzepte, wt Werkstatttechnik online H.3, 2003, pp. 17-181.
14. Mäntylä M, Nau D, Shah J. Challenges in Feature-Based Manufacturing Research, Communication of the ACM February 1996/Vol.39 No. 2, 1996, pp. 77-85.
15. Schiller E, Seuffert WP. Digitale Fabrik bei DaimlerChrysler, Sonderdruck Automobil Produktion 2/2002, pp. 4-10 (in German).
16. Thoben KD, Weber F. Information and Communication Structures for product development in the Concurrent Enterprise: Requirements and Concepts, Life-Cycle Approaches to Production Systems- Management, Control, Supervision, the Annual Conference of ICIMS-NOE, Budapest, 1997, pp. 460-467.
17. VDI, VDI-Richtlinie 2218: Feature-Technologie, (in German), 1999.
18. VDI, VDI-Richtlinie 4499: Digitale Fabrik, (in German), Düsseldorf, 2006.
19. Weber C. What is a Feature and What is its Use? – Result of FEMEX Working Group I, Proceeding of the 29th International Symposium on Automotive Technology and Automation, Florence, Italy, 1996, pp. 109-116.
20. Westkämper E, Jovanoski D, Rist T. New Framework for Digital Factory Planning, Proceedings of the 36th CIRP International Seminar on Manufacturing Systems, Saarbrücken, Germany, 2003, pp. 191-198.
21. Wiendahl HP. Auf dem Weg zur „Digitalen Fabrik", wt Werkstattstechnik 4 (in German), 2002, p. 121.