AN APPROACH TOWARDS THE STRUCTURING OF METHODS – THE BASIS FOR A FLEXIBLE METHOD-TOOL

Stephan Walter

Product Development Machine Elements pmd
University of Technology Darmstadt
Magdalenenstrasse 4

64289 Darmstadt
GERMANY
E-mail: walter@pmd.tu-darmstadt.de

Herbert Birkhofer

Product Development Machine Elements pmd
University of Technology Darmstadt
Magdalenenstrasse 4

64289 Darmstadt
GERMANY
E-mail: birkhofer@pmd.tu-darmstadt.de

Computer-aided learning, knowledge-based training, knowledge-based systems, learning environments

Abstract

This paper describes an approach towards the structuring of product development methods. This is necessary in order to build up method-tools according to these structures, which support both the designer in practice and the student at the university.

The product development methods have to be structured in such a way that their similarities become clear. Different structuring criteria are pointed out with their properties and the dependencies among them.

In further steps, the requirements on the method-tools and the circumstances of their application are shown. In conclusion, the following (future) work steps toward developing method-tools are described.

1 Introduction

Nowadays the product developers need efficient support due to quickly changing global markets and decreasing time-to-market.

Often problems in method application are caused by changed circumstances in product development, limits and possibilities of methods and consequential errors, inadequate customized methods concerning specific circumstances or absence of adapted schooling. [Zanker, W., 2000] An effective and efficient support of the daily work of product developers will be enabled primarily through the use of computer support. Many product development methods are more useful and their application will be enforced by computer support.

But to prevent product developers from using a specific method-tool for each product development method they want to apply, the aim should be to establish only a few flexibly configurable method-tools, which can be adapted to several applications and situations, as well as to different product development methods.

To achieve this adjustability of the method-tool, it is useful to configurate its architecture similar to the structures of product development methods. A likely question hereby: Which types of structures are enclosed in product development methods?

2 Aims and objectives

The aims of software-support for product development should be to aid both education in product development and the daily work of product developers in industry.

This challenging goal shall be reached by a teaching and learning system named pinngate at Darmstadt University of Technology. The system will contain the following objects: basic solutions, modularized content, didactic concepts, and method-tools.

The interfaces between the method-tools and the other pinngate-components are described as follows (Figure 1):

- Interface to the basic solutions [Sauer, T., Kloberdanz, H., Walter, S., Berger, B., Birkhofer, H., 2003]:

  The existing basic solutions and solution descriptions will be used by the method-tools as a kind of input in order to be processed by them.

  This means that the method-tools will use the solution descriptions in order to generate new solutions to given design tasks; however, they are forced to define the newly found solutions along the given structure of properties of the solution descriptions. The attributes and values of the structure properties of the solution descriptions are the components, which are processed by the method-tools.

  Therefore, the properties of the basic solutions and the structures of the methods will have to be arranged.

- Interface to the modularized contents [Birkhofer, H., Berger, B., Walter, S., 2002]:

  The modularized contents shall assist the users with explanations about how to proceed and how to use the method-tool. Furthermore, hints should be given to the user in order to guide his method application.

  To enable an easy-to-handle assistance by such hints and explanations, it could be of advantage to develop the method-tools following a standardized structure common to each method-tool.

- Interface to the didactic concepts [Jänsch, J., Sauer, T., Walter, S., Birkhofer, H., 2003]:

  Requirements are given for the adaptability of the method-tools concerning the fulfillment of the didactic concepts. Because of the situation-specific circumstances of the method application in the didactic concept, the method-tool will have to be adaptable to these special needs.

  For example, a didactic concept to impart knowledge about product development methods to students in lectures or exercise courses at university will be developed different from one that imparts the same product development methods in a workshop with practicing designers in industry. The difference will not only be seen in the restriction while processing, but also in the given support. Students as well as inexperienced / amateur users will need more guidance (the method-tool should be more chain-driven) than practiced users (the method-tool should be more event-driven) who wish to integrate their specific experiences.

  This adaptation to the user- and situation-specific needs shall be realized by trimming parameters, which strongly depend on the properties of the structuring criteria of the product development methods.
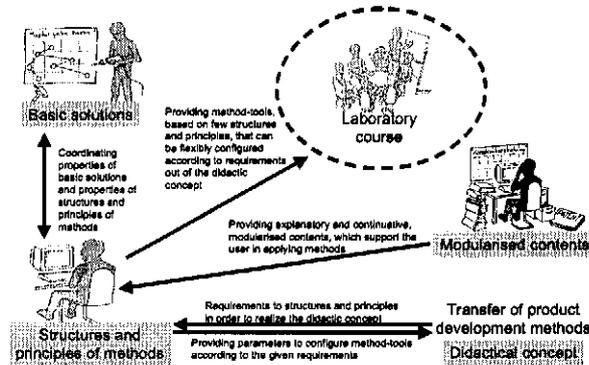
Figure 1: Interfaces between the pinngate-components

From the flexible cooperation of the single components – basic solutions, modularized contents, structures and principles of methods, and didactical concept – a system arises, which will be adaptable and adjustable to the different demands of potential users.

So, one benefits from the cooperation of the single components in contrast to their individual usage.

This paper will now focus on the method-tools, especially their structures and working principles.

"The aim is to implement product development methods into software-tools and to do that, above all, with as little expense as possible. To reach that aim the methods should be structured or even standardized described. This would mean that no product development method would have to be transferred to an extra software-tool. The specific software-tools could be put together from only a few 'standard-elements' and could be flexibly configured." [Walter, S., Sauer, T., Weiss, S., Birkhofer, H., 2003]

Above all, the requirement on the software support to ensure the adjustability and to fulfill the necessity of adjustment by the didactic concept causes an exactly descriptions of the properties and structures of the product development methods.

In order to build up flexible method-tools, which can meet most of the given requirements of the didactic concept, and also collaborate with the basic solutions, product development methods will have to be analyzed and described with their properties and attributes.

3 An approach to the structuring of methods

Different criteria may be used in order to structure product development methods. The following enumeration gives a wide, yet incomplete, range of possible criteria. The structuring criteria are not intended to select product development methods for a specific design task, but to describe the method itself. Therefore the following criteria have to seen under this aspect.

The specific structuring criteria are not unconditionally independent from each other. Theoretically, some values of structuring criterion A could seriously constrain the range of values of structuring criterion B.

Some of the given structures are, depending on their properties, more or less convenient for structuring. This consideration will have to be made in future steps.

3.1 Type of input data

The type of input data could be used as structuring criterion for methods. Possible values for these types of input data are graphics, text, and digits.

This structuring criterion is a purely formal criterion, since no information is given about the content of the data to be processed.

The type of processing as well as the type of output data will most probably be influenced by the type of input data. The output data will be equal to the input data, if the processing method has an exclusively structuring character. In that case, the data will be just resorted or arranged in a given scheme.

The expense of processing the data by the method depends on the type of input data. The processing of text in an editor or digits in a calculator is much less expensive in comparison to the processing of graphical data in picture editing programs or CAD programs.
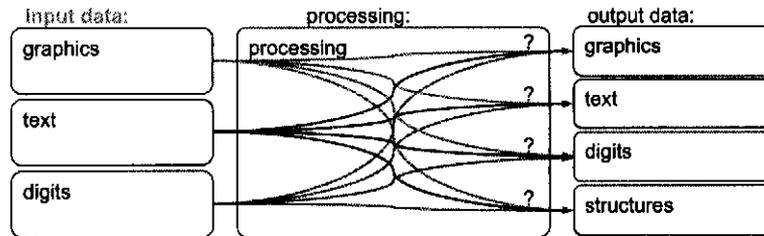
Figure 2: Type of input data

3.2 Type of output data

The type of output data also is a purely formal structuring criterion. Possible values of output data are text, digits, graphics, and structures (order, succession).

With methods of exclusively structuring character, the output data will not differ from the input data after processing.

The type of output data has an immense influence on the processing effort. Similar to the influence of the input data on the processing, the effort increases from the output of text to the output of graphical elements.
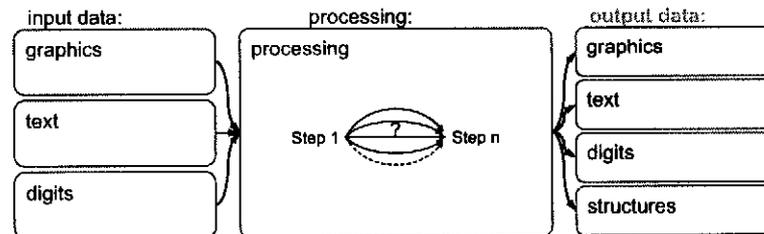
Figure 3: Type of output data

3.3 Type of application

Furthermore, methods can be structured by taking a look at the type of the user-group, which is necessary for the method application, respectively the moderation of the user-group.

It may be differentiated between 'single-user-methods' (e. g. the generation of a requirements list or the usage of a check-list) and 'multiple-user-methods' respectively 'group-methods' (e. g. the performance of a brainstorming or brain-writing workshop). In addition to that, methods may differ in the point of needing moderation while applying the method (e. g. the performance of a brainstorming) or not (e. g. the usage of a check-list).
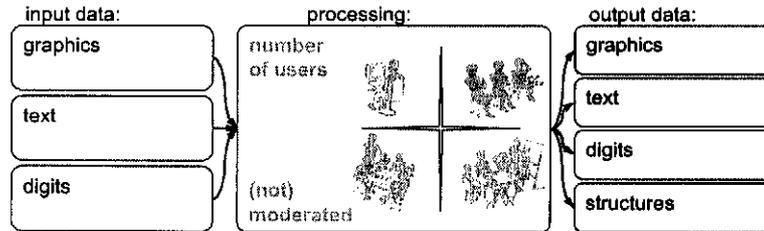

Figure 4: Type of application

3.4 Structure of user-group

Another criterion to structure methods, which depends on the users as individuals, is the structure of the user-group. There are methods that can be applied exclusively by technical experts, such as a failure mode and effect analysis, for example. Contrary to this, there are methods, especially those, which are conducive to creativity and help, find new solutions to tasks that require at least some beginners in the user-group. These beginners fulfill the part of lateral thinkers, which help the team to leave the beaten tracks.

Furthermore there is an impact on the method if the users in the group know each other, respectively not. [Wulf, 2001]

This structuring criterion sometimes corresponds to the moderation of the method. Eventually, it could be necessary that a technical expert moderate a group of beginners, in order to discover new goal-oriented ideas for a task.
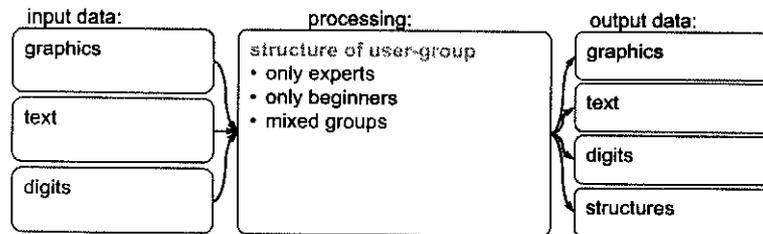

Figure 5: Structure of user-group

3.5 Restrictions while processing

The structuring criterion 'restrictions while processing' could be divided into the two cases "event-driven" and "chain-driven".

Chain-driven means that the users' worksteps will be routed by the method.

Event-driven means that the proceeding can be chosen by the user – with certain limitations.

232

According to this, the structuring of methods could take place corresponding to more leading method proceedings, which are primarily useful to users who are rather inexperienced in method application. Also a structuring of methods could occur corresponding to more open method proceedings, which are useful to users who know how to plan their worksteps. These users have the experience to know which data has to be provided at a given point.

Depending on the method, these structures are more or less in high gear. For example, an evaluation method is more chain-driven: there is a logical procedure, which first deduces evaluation criteria and edits and/or completes them, and then the existing solutions are evaluated in respect to the assessed evaluation criteria. Finally, the results of the evaluation are displayed in a strength diagram, for example. This logical procedure does not leave the user much room for expansion.

In contrast to that procedure, the generation of a requirements list provides the user with more freedom. Whether first collecting requirements and classifying them later, or collecting requirements along product lifecycle phases, it does not make a big difference concerning the result of the method application.
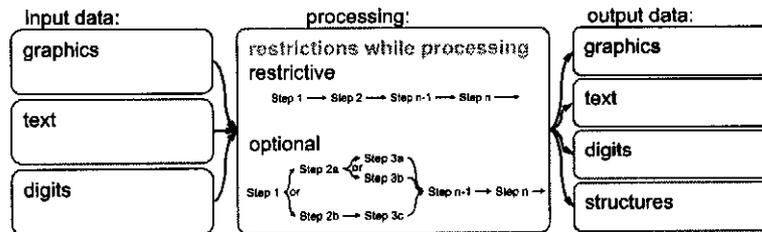

Figure 6: Restrictions while processing

3.6 Type of presentation

Another purely formal structuring criterion is the type of presentation. Possible forms are tables, graphic structures or models, or simple text.

At a first glance, there is no difference to the processing of the method-tool whether the user enters all of the input data in a clear and complete table, which allows him to get an overview of all the data he has to enter, or if the method-tool leads him through the procedure by asking the user to enter individual details of data piece by piece.

But it makes a significant difference to the user, if he does not receive an overview of what the entered data is for. So, it could become more difficult to apply the method.
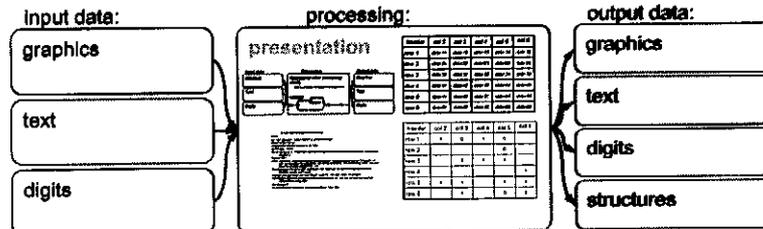

Figure 7: Type of presentation

3.7 Type of processing

The type of processing of the method differs between 'generation of new contents / data', 'calculation of data', 'structuring, respectively arranging of data', 'assembling of data', and combinations of the given types (e. g. an evaluation is composed of the calculation of the rating and arranging the evaluated solutions along the calculation).

Purely structuring methods do not handle the input data. The output data (method result) is equal to the input. But the information 'between the data' is the real result of the method application. The processing of the method is not a type of processing but simple arranging.

The processing is the real help to the user. The product development method itself is not creative – it just can support the user. The processing inside the method generates the results, which are the added value of the method application.

The work steps of the product development method take place along algorithms and / or rules. These should be repeatable and reusable in different method-tools.
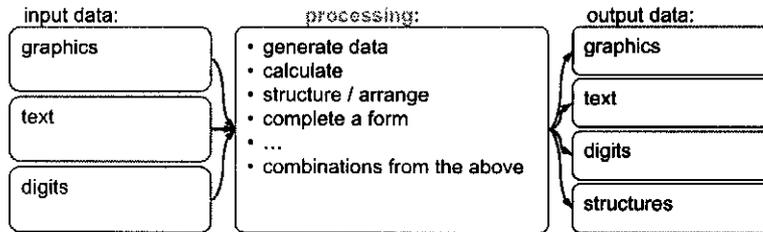
input data:	processing:	output data:
graphics	• generate data • calculate • structure / arrange • complete a form • ... • combinations from the above	graphics
text		text
digits		digits
		structures

Figure 8: Type of processing

3.8 Preparation of processing

Another way to structure and to distinguish methods from each other is the kind of how their application has to be prepared. Some product development methods are self-explanatory or at least understandable with a short introduction. Others are only applicable by studying detailed documents. For example, failure mode and effect analyses and quality function deployments are very complex product development methods. They are not feasible without an extensive preparation.

This structuring criterion also depends on the existence and kind of possible moderation.

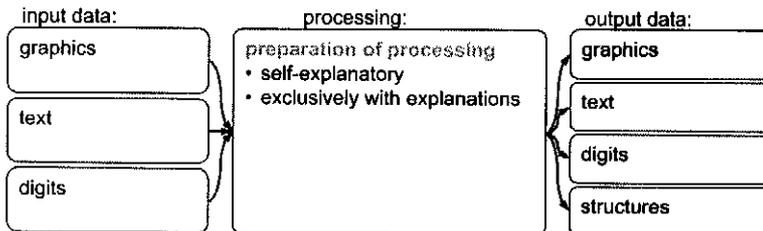| input data:	processing:	output data:
graphics	preparation of processing • self-explanatory • exclusively with explanations	graphics
text		text
digits		digits
		structures

Figure 9: Type of preparation of processing

3.9 Duration of preparation

The duration of preparation for the product development method application strongly depends on point 3.8, the type, and the expense of the preparation.

Also, the connection with the point 3.10 "Duration of processing" exists in most cases. An extensive preparation is mostly reflected in an extensive procedure and even an extensive duration of processing.

In general, the differentiation could be made between 'spontaneously applicable methods' and 'only with leadtime / preparation applicable methods'.

In many cases, the duration of preparation depends on the range of inquiry that is necessary for the application on the method (e. g. to prepare the application of an evaluation), or on the effort to generate the procedure documentation.
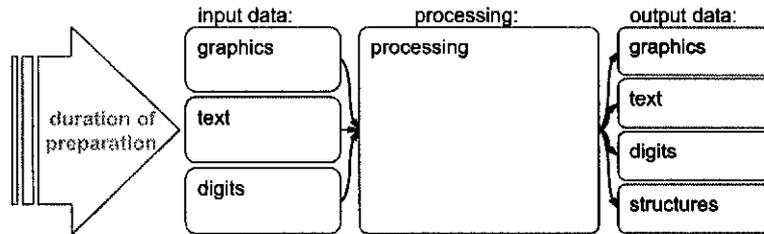

Figure 10: Duration of preparation

3.10 Duration of processing

Structuring product development methods by the duration of processing could be done in the steps short term, middle term, and long term. Short term may be up to one hour (e. g. for a brain-writing / method 635), middle term up to one day (e. g. an evaluation), and long term more than one day (e. g. for a failure mode and effect analysis).
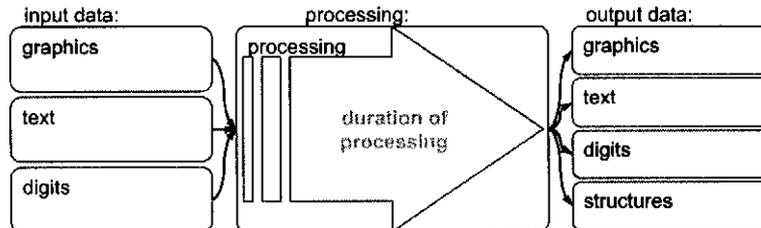

Figure 11: Duration of processing

3.11 Duration of analysis

The duration, which is necessary to analyze the results, is a structuring criterion concerning the availability of the results of the method application.

The question is how far the results of method applications are directly available and usable for the following worksteps. It is possible that they first have to be processed by another method to make them usable. The answer to this question is important to the duration of the analysis.

An evaluation, for example, generates an order of rated solutions according to their quality to fulfill given requirements. Although the results will most likely leave potential for discussion, a first ranking is established.

In contrast to this, the results of a brainstorming or brain-writing workshop, for example, will have to be reviewed in order to determine possible duplicates. Therefore, these results are not directly usable.
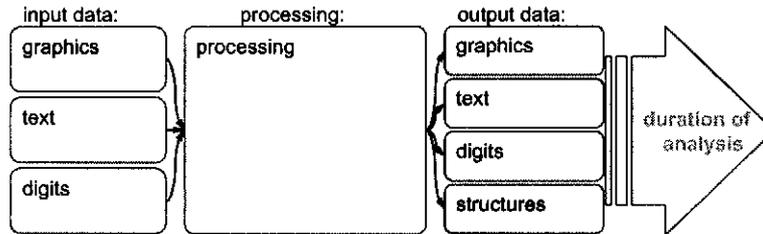

Figure 12: Duration of analysis

4 Conclusion

The starting point for the analysis of product development methods might not only be the type of representation – as assumed in [Walter, S., Sauer, T., Weiss, S., Birkhofer, H., 2003]. Furthermore, other structuring properties of product development methods, such as those given above, must be considered.

The problem to be solved is how to eliminate the conflict of interests between the needs (from the didactic concept) and the possibilities to adjust depends on structures and principles of the product development methods (Figure 13).
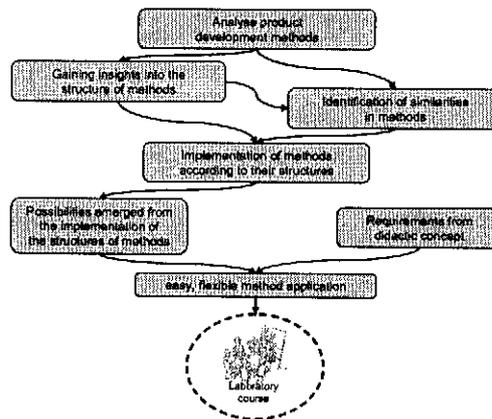

Figure 13: Effects on the method-tools, which result of the structure of methods and of the requirements of the didactic concept

By analyzing product development methods the structuring criteria of these are described with their properties and attributes. The insights into the structure of product development methods

gained through this analysis will help to identify the similarities of methods. These similarities will not only be the result of the structuring criteria, but will also be influenced by the method itself.

The implementation of product development methods as method-tool(s) according to their structures will reveal possibilities to fulfill the requirements of the given didactic concepts. In conclusion, an easy to use and flexible method application shall be possible.

5 Contribution

The aim to support designers at work as well as students in university shall be fulfilled by flexible method-tool(s). To build up these tools it is necessary to know the structures they should have. As they are to be adapted to several situations and tasks, as well as to different product development methods, they ought to be of a flexible structure. Method-tool(s) that are adaptable to represent different product design methods simplify the use of product development methods for the users. So the method-tool(s) will have an effective and efficient impact on the daily work of product developers.

6 Outlook

The next steps will have to bring out the most significant structuring criteria. Their interrelations have to be examined. Based on these cognitions, method-tool(s) can be developed and tested considering the requirements of the pinngate-system.

In tests in laboratory courses, first experiences with these method-tools will take place (Figure 1). Supported by adapted method-tools, small student groups will handle manageable tasks. In this way, it will be tested whether the adjusting parameters are the right ones. Furthermore, it has to be clarified whether the method structures and principles allow a window of opportunity to comply with the requirements given by the didactic concept.

The experiences gained will be useful in the implementation of further product development methods.

References

Birkhofer, H., Berger, B., Walter, S., "Modularization of Knowledge – A New Approach in the Field of Product Innovation", in Proceedings of the 7th International Design Conference DESIGN 2002, Dubrovnik 2002, Vol. 1, pp. 289 – 294.

Helbig, D., "Entwicklung produkt- und unternehmensorientierter Konstruktionsleitsysteme", Dissertation, Institut für Maschinenkonstruktion, TU Berlin, 1994.

Jänsch, J., Sauer, T., Walter, S., Birkhofer, H., "User-suitable Transfer of Design Methods", in Proceedings of the 14th International Conference on Engineering Design ICED 2003, Stockholm 2003, Vol. Research for Practice, pp. 371 – 372.

Pahl, G., Beitz, W., "Engineering Design", 2nd Edition, Springer Publisher London, 1996.

Sauer, T., Kloberdanz, H., Walter, S., Berger, B., Birkhofer, H., "Describing Solutions for the Conceptual Phase – Structured and User-oriented", in Proceedings of the 14th International Conference on Engineering Design ICED 2003, Stockholm 2003, Vol. Research for Practice, pp. 49 – 50.

Walter, S., Sauer, T., Weiss, S., Birkhofer, H., "Efficient Support for the Product Development Process – Computer-based Method-tools for the Conceptual Phase", in Proceedings of the 14th International Conference on Engineering Design ICED 2003, Stockholm 2003, Vol. Research for Practice, pp. 265 – 266.

Wulf, J. E., "Elementarmethoden zur Lösungssuche", Dissertation, Lehrstuhl für Produktentwicklung, TU München, 2001.

Zanker, W., "Situative Anpassung und Neukombination von Entwicklungsmethoden", Dissertation, Lehrstuhl für Produktentwicklung, TU München, 2000.