Proceedings of the 4th WDK Workshop
on Product Structuring
October 22-23, 1998
Delft University of Technology
Delft, The Netherlands

Specifying Relationships in Product Families to enable STEP-based Implementations

Alison McKay and Alan de Pennington
School of Mechanical Engineering, University of Leeds, Leeds LS2 9JT, United Kingdom

Abstract

If companies are to gain advantage from adopting STEP (STandard for the Exchange of Product model data [ISO10303]) then the gap between the application context of an Application Protocol (AP) and the requirements of the engineering process chain must be bridged. This paper illustrates a way in which general purpose data models, such as those in a STEP Application Protocol, can be used to support the definition of product families.

1 Introduction

Many companies are striving to exploit the potential of product data technology. The emerging STEP standard [1] is today's most significant manifestation of product data technology. Key aspects of STEP's data specification architecture include the Integrated Resources, which specify fundamental concepts and relationships that occur in product data in general, and Application Protocols, which define usages of the Integrated Resources to support the requirements of specific application contexts. In both cases descriptions of individual products are supported. The engineering processes of many companies, though, are built around company-specific product families. Thus, if the STEP standard is to be exploited to the full by such companies then a general purpose mechanism for relating product data defined through product families to product data specifications that are intended to support the description of individual products is required. This paper proposes such a mechanism.

Earlier research on relating product family data to general purpose product data resulted in a clearer understanding of the problem of linking general purpose product data models with more specific ones [2]. This paper reports on the use of a prototypical data specification language in specifying relationships between general purpose and more specific product data models. The language is helping in understanding such relationships. An office chair product family is used as an example of the specific and a STEP-like data model is used as an example of the general purpose. Our goal is to help make STEP usable as well as implementable by industry. To be usable STEP must fit into existing processes and the languages therein, for example, the language of specific product families.

2. Case Study: Office chair realisation process

The case study used in this paper is presented as a process supported by engineering software tools (applications). From this basis, the information used by such applications in the context of the process is expanded.

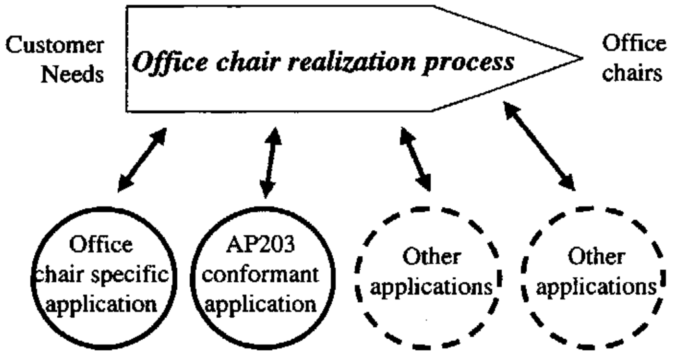


Figure 1: Example Process

Figure 1 shows an example realisation process for a product that is a member of an office chair product family. It starts with the customer needs and ends with the product - office chairs. The process is supported by applications. Some applications are general purpose software tools, for example AP203 [3] conformant applications, whilst others are more specialised, for example, office chair specific applications. All of these applications must be usable from within the realisation process if the process itself is to benefit. A problem for companies is that it is not easy to build interfaces between product family specific processes and general purpose software tools. This is due in part to a mismatch between the information structures that underpin the process and those that underpin the software tools that are to be used. Since the information requirements of the applications govern the format of STEP APs to which they will conform, this explains why there is often a need to be able to build upon general purpose STEP APs to allow them to be tailored to the information needs of specific engineering processes.

3. Problem Statement

As stated above, STEP APs are general purpose product data exchange standards. However, many engineering processes are tailored to the needs of specific families of products. STEP does not (and probably should not) standardise product family definitions. However, STEP (and other general purpose tools) does need to be used in contexts that are specialised, preferably without losing their generality. This paper describes a mechanism that is built upon the idea of levels of instance [4]. The long term aim is to keep both generality and speciality in product data and to relate them in a coherent, computer-sensible way. This is based on the premise that the answers to the questions "Should STEP standardise family definitions?", and "Should companies use more general language?" are both "no".

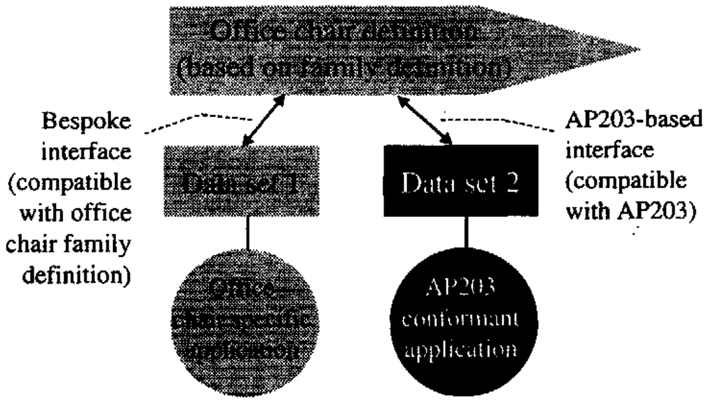


Figure 2: The role of APs

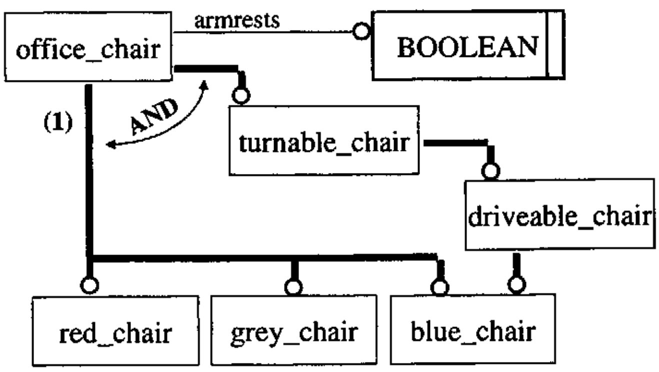
The role of APs is illustrated by Figure 2. It can be seen that the format of the data exchanged between application and process is governed by the application. A disadvantage of family-specific applications is that their interfaces are bespoke whilst an advantage is that they fit well within the process. On the other hand, interfaces to general purpose applications can be standardised but they do not fit well into the process. This paper is targeted at providing a better fit between STEP APs, and so applications that conform to them, and the processes that they are intended to support.

4 Case Study: Information Requirements

In this section the way in which applications are linked to a process, with a focus on the format of the data that they use, is considered. Two data formats are presented. The first is an example of the sort of data elements and structure that one would find in the Data Set 1 box on Figure 2. The second is an equivalent example of Data Set 2. In both cases a data model (written using EXPRESS-G [5]) and an example instance (written using EXPRESS-I-G [6]) are presented.

4.1 Office chair family information requirements

An example data model which shows a data format for descriptions of office chairs (Data set 1 on Figure 2) is given in Figure 3.



NOTE: blue chairs are driveable, driveable chairs are turnable

Figure 3: Office chair data format

It can be seen that the concepts embodied by this data model are specific to office chairs and would be of little use in other contexts.

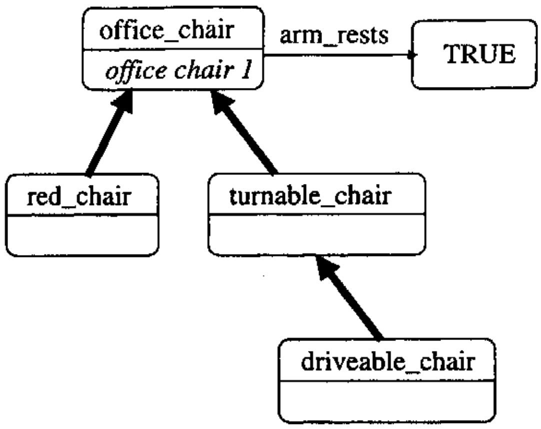


Figure 4: Example definition of an office chair

Figure 4 shows an example definition of an office chair defined in terms of the office chair data model in Figure 3. It is a red driveable chair with armrests. It can again be seen that the concepts therein are specific to the office chair product family.

4.2 General purpose product information requirements

A data model that allows product data that relates to many different kinds of product (Data set 2 on Figure 2) is given in Figure 5. This data model is based upon data elements from the STEP GPDR (Generic Product Description Resource) [7] that typically occur in STEP APs. It allows product structure to be specified.

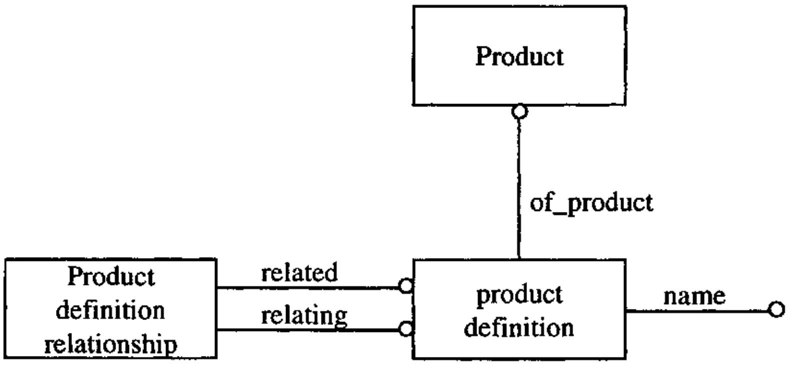


Figure 5: General purpose (STEP-like) format

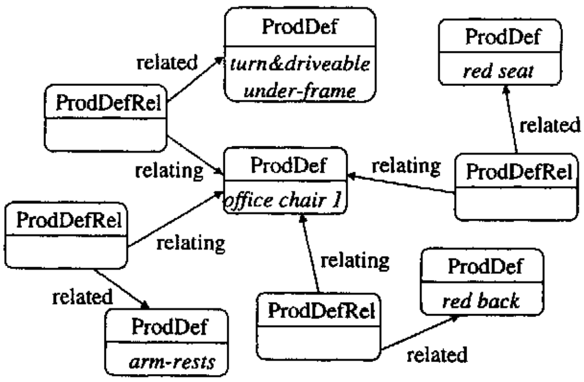


Figure 6: Example definition of an office chair

In Figure 6 an instance of the data model given in Figure 5 is shown. As in Figure 4, the chair is red and driveable with armrests. However, unlike Figure 4, the format of the data is not bound to the office chair product family. The product structure of many kinds of product, not just office chairs, could be defined in this format.

To aid understanding of Figure 6, the product structure that is represented in Figure 6 is shown in Figure 7.

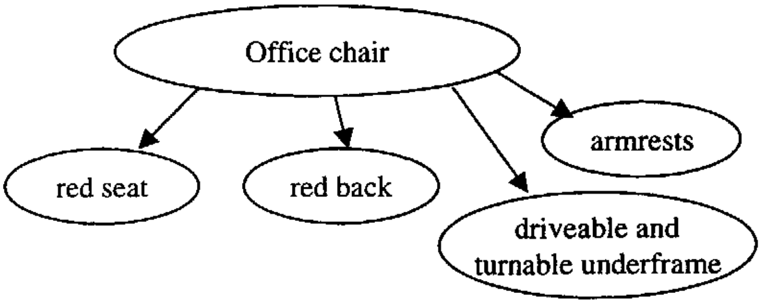


Figure 7: Office chair product structure

The ovals in Figure 7 represent products and the arrows represented decomposition relationships. Thus it can be said that the example office chair is composed of a red seat, a red back, a driveable and turnable underframe and armrests.

5 Levels of Instance

The problem that this paper addresses is in relating these dissimilar definitions of the same product.

Levels of instance provides a general purpose mechanism to link the general with the specific. It is based on the assertion that, if the general purpose data model is truly general and includes the context of the family, then it should be possible to define the family and its members in terms of (as an instance of) the general purpose data model. LOIL-G is the graphical notation of a prototypical data specification language (LOIL) for the specification of the structural aspects of levels of instance [8]. LOIL has roots in the EXPRESS data specification language of STEP [5] but is not constrained to what is essentially a two-layer instance/data model architecture.

LOIL definitions of the data models given in Figures 3 and 5, and of instance relationships between them, are given in the appendix to this paper. A formal syntax for LOIL exists and the lexical version of the data models are given in the appendix. Figure 8 shows how LOIL-G can be used to relate the instances and data models introduced in this paper. In this way a specification of the relationship between the general and the specific office chair has been demonstrated. At this stage LOIL only supports the structural elements of the relationship. For implementation, mechanisms to facilitate the correct options of selection concepts in intermediate layers are required. An early experimental implementation in the Leeds Structure Editor [9] uses case statements for this purpose.

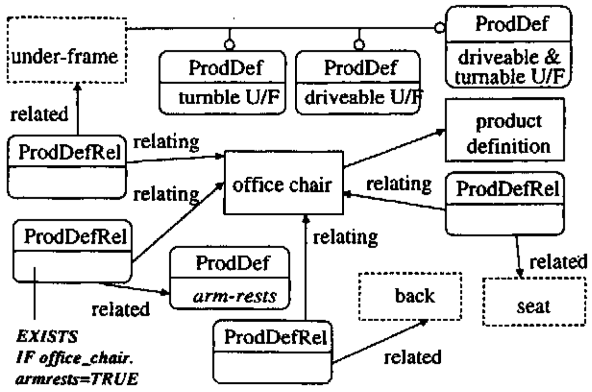


Figure 8: Relating the specific office chair definition to the specific

6 Concluding Remarks

Levels of instance can be used to specify relationships between general purpose and more specialised product data models. "Partial instantiation" is a data modelling abstraction that allows levels of instance to be defined. LOIL (a prototype language that supports partial instantiation) has been demonstrated as a means of relating a general purpose (STEP) data model with a specific (product family) data model. This mechanism has the potential for allowing STEP-based implementations of product family data. However, this paper reports research at an early stage of development and significant effort will be needed to create implemented solutions.

The need to be able to tailor STEP APs to suit the requirements of specific engineering processes has been identified by Vaughan [10]. Within the STEP community effort is being directed towards adapting the STEP data specification architecture [11] to address this need. In this paper a means of relating general purpose product data models with more specific ones has been introduced. It has been presented as a means of tailoring STEP APs to suit specific processes. However, this is a general purpose mechanism which, by providing the beginnings of a formally define language to relate the general and the specific, allows one contributor to complexity in product data to be addressed.

References

[1] ISO/DIS 10303-1 - Industrial Automation Systems and Integration - Product Data Representation and Exchange - Overview and fundamental principles, December 1994. International Standard.

[2] Alison McKay, Frederick Erens, and M Susan Bloor. Relating product specifications and product definitions in a product model. *Research in Engineering Design,* 2:63 - 80, 1996.

[3] ISO 10303-203 - Industrial Automation Systems and Integration - Product Data Representation and Exchange - Application Protocol: Configuration controlled design, December 1994.

[4] Alison McKay, M Susan Bloor, and Alan de Pennington. Levels of Instance in Product Models. In *23rd International Symposium on Automotive Technology and Automation, volume II,* pages 102 - 109, December 1990.

[5] ISO 10303-11 - Industrial Automation Systems and Integration - Product Data Representation and Exchange - Description Methods: EXPRESS Language Reference Manual, December 1994.

[6] Alison McKay, Frederick Erens, M Susan Bloor, and Jon Owen. EXPRESS-I-G: A graphical notation for EXPRESS-I. *Third Annual EXPRESS Users' Group - EUG'93,* October 1993. Berlin, Germany.

[7] ISO 10303-41 - Industrial Automation Systems and Integration - Product Data Representation and Exchange - Integrated Generic Resources: Fundamentals of Product Description and Support, December 1994.

[8] A McKay. *Relationships in product data.* PhD thesis, Department of Mechanical Engineering, University of Leeds, Leeds, UK, September 1996.

[9] Frederick Erens, Alison McKay, and Susan Bloor. Modelling Multiple Levels of Abstraction: Instances as Types. *Computers in Industry,* 24(1):17-28, May 1994.

[10] Chris Vaughan. Industrial framework model, December 1998. Available from `http://wg10step.aticorp.org/Deliverables/index.html`.

[11] William F Danner, David T Sanford, and Yuhwei Yang. STEP Resource Integration: Semantic and Syntactic Rules, January 1992. ISO TC184/SC4/WG5 N10 (NISTIR 4528), Available from ISO TC184/SC4 Secretariat, National Institute of Standards and Technology, Gaithersburg, MD20899, USA.

APPENDIX

```
SCHEMA  office_chair_schema ;

  REFERENCE FROM  STEP_like_schema ;

  CONCEPT  chair_back INSTANCE_OF  product_definition ;
    SELECT (  red_back ,  grey_back ,  blue_back )
  END_CONCEPT ;

  CONCEPT  back_in_chair INSTANCE_OF  product_definition_relationship ;
  VALUES
    chair:  product_definition_relationship. relating =  office_chair;
    back:  product_definition_relationship. related =  chair_back;
  END_CONCEPT ;

  CONCEPT  chair_seat INSTANCE_OF  product_definition ;
    SELECT (  red_seat ,  grey_seat ,  blue_seat )
  END_CONCEPT ;

  CONCEPT  seat_in_chair INSTANCE_OF  product_definition ;
  VALUES
    chair:  product_definition_relationship. relating =  office_chair;
    seat:  product_definition_relationship. related =  chair_seat;
  END_CONCEPT ;

  CONCEPT  turnable_underframe INSTANCE_OF  product_definition ;
  VALUES
    name_value:  product_definition.name =  "turnable_underframe"; -
    of_value:  product_definition.of_product =  turnable_underframe_product;
  END_CONCEPT ;

  CONCEPT  driveable_underframe INSTANCE_OF  product_definition ;
  VALUES
    name_value:  product_definition. name =  "driveable_underframe";
    of_value:  product_definition. of_product =  driveable_underframe_product;
  END_CONCEPT ;

  CONCEPT  driveable&turnable_underframe INSTANCE_OF  product_definition ;
  VALUES
    name_value:  product_definition. name =  "driveable&turnable_underframe";
    of_value:  product_definition. of_product =  driveable&turnable_underframe_product;
  END_CONCEPT ;
CONCEPT  underframe INSTANCE_OF  product_definition ;
    SELECT    (          turnable_underframe_product      ,          driveable_underframe_product     ,
driveable&turnable_underframe_product )
  END_CONCEPT ;

  CONCEPT  underframe_in_chair INSTANCE_OF  product_definition_relationship ;
  VALUES
    chair:  product_definition_relationship. relating =  office_chair;
    underframe:  product_definition_relationship. related =  underframe;
  END_CONCEPT ;

  CONCEPT  armrests_in_chair INSTANCE_OF  product_definition_relationship ;
  VALUES
    chair:  product_definition_relationship. relating =  office_chair;
    armrests:  product_definition_relationship. related =  armrests;
  END_CONCEPT ;

  CONCEPT  armrests INSTANCE_OF  product_definition ;
  VALUES
    name_value:  product_definition. name =  "armrests(as_designed)";
    of_value:  product_definition. of_product =  armrests_product;
  END_CONCEPT ;

  CONCEPT  armrests_product INSTANCE_OF  product ;
  VALUES
    name_value:  product. name =  "armrests";
  END_CONCEPT ;

  CONCEPT  office_chair INSTANCE_OF  product_definition ;
  SUPERTYPE OF ( AND ( ONE_OF (  red_chair ,  grey_chair ,  blue_chair )  ,  turnable_chair ) ) ;
  VARIABLES
    armrests :  BOOLEAN ;
    id       :  NAME ;
  VALUES
    name_value:  product_definition. name =  "office_chair";
    of_value:  product_definition. of_product =  office_chair_product;
  END_CONCEPT ;

  CONCEPT  office_chair_product INSTANCE_OF  product ;
  VALUES
    name_value:  product. name =  "office_chair";
  END_CONCEPT ;

  CONCEPT  turnable_chair INSTANCE_OF  product_definition ;
  SUBTYPE OF (  office_chair ) ;
  SUPERTYPE OF (  driveable_chair ) ;
    ??
  END_CONCEPT ;

  CONCEPT  driveable_chair INSTANCE_OF  product_definition ;
  SUBTYPE OF (  turnable_chair ) ;
  SUPERTYPE OF (  blue_chair ) ;
   ??
  END_CONCEPT ;

  CONCEPT  red_chair INSTANCE_OF  product_definition ;
  SUBTYPE OF (  office_chair ) ;
    ??
  END_CONCEPT ;

  CONCEPT  grey_chair INSTANCE_OF  product_definition ;
  SUBTYPE OF (  office_chair ) ;
     ??
  END_CONCEPT ;

  CONCEPT  blue_chair INSTANCE_OF  product_definition ;
  SUBTYPE OF (  office_chair ,  driveable_chair ) ;
    ??
  END_CONCEPT ;

END_SCHEMA ;

SCHEMA  STEP_like_schema ;

  CONCEPT  product INSTANCE_OF  concept ;
  VARIABLES
    name :  NAME ;
  END_CONCEPT ;

  CONCEPT  product_definition INSTANCE_OF  concept ;
  VARIABLES
    name        :  NAME ;
    of_product  :  product ;
  END_CONCEPT ;

  CONCEPT  product_definition_relationship INSTANCE_OF  concept ;
  VARIABLES
    related   :  product_definition ;
    relating  :  product_definition ;
  END_CONCEPT ;

END_SCHEMA ;
```

4th WDK Workshop on Product Structuring
October 22-23, 1998
Delft University of Technology
Delft, The Netherlands

METRICS FOR SUPPORTING THE USE OF MODULARISATION IN IPD

Asko Riitahuhta, Antti Pulkkinen, Mogens Myrup Andreasen

Modularisation stages, Metrics for Modularisation, Dynamic Modularisation

Abstract

In this paper an approach to measure modularisation process is offered. The approach is based on the inherent relation between modular engineering and its' business imperatives. Also some other metrics found in literature are presented briefly. The product architecturing principles are presented. Product quality aspects are described in conjunction to modularisation and modularity. The concept of modularisation stage is introduced and some indications of the different stages are presented as an example. A stakeholder analysis for evaluating the central concepts; profitability, growth, and knowledge potential, of modularisation is proposed. Analysing metrics the life cycle of modular systems against needed dynamism are described. Using the stage-gate and the here introduced Modularisation Implementation Profile (MIP) methods for management of modularisation process are suggest.

1. Introduction

Measuring modularisation is relatively new subject. Because modularisation is gaining more importance with the Post Mass Production and Mass Customisation Paradigms, it is necessary to create measurement concepts and systems for it. In this paper some existing methods found from literature are presented. Based on the literature review and experiences in industry, we see that there are three product architecturing principles. Then we present a theory base for metrics; its' business relations and the quality aspects in relation to modularisation. We also present the concept of modularisation stage and decision support methods (stakeholder analysis and measuring concepts for life cycle of modular systems) for analysing how to adopt modularisation stages: to stick to an existing stage or to proceed to another stage? We also present Modularisation Implementation Profile (MIP) charts for measuring the effectiveness of modularisation project and suggest stage gate method for managing it. The presented methods are easy to computerise (e.g. by using a spreadsheet program).

1.1 Main problems in Modularisation metrics

Modularisation is an effective means for structuring a product or in fact a product family, because modularisation enables the creation of a group of end products that are at the same time alike but different. From another point of view Modular Engineering (ME) is a philosophy and a methodology, which creates possibilities in obtaining business results. Rationalising marketing, product development, production, and sales achieve these results. As a consequence it is ambiguous, what shall be covered by ME metrics. Is the issue to measure the internal or the external efforts and effects of Modular Engineering? Should we put the focus on the rationalisation effects of modularisation and how direct should be the relation between Modular Engineering and the induced effects in operation?

1.2 Modular Engineering and business

Today it is well known fact that in a product development project choice of technical solutions and product design in general have a decisive influence on the company's competitive edge. Design has an effect on the conditions in the company's operation. These operations (marketing and sales, production, storage, etc.) happen in the internal life cycle phases of a product. Product design also has an effect to the external life cycle phases, where the operations like usage, maintenance, and re-cycling occur.

Thus, it is a good way to enhance the life cycle phases and to rationalise business operations already in product design. This insight is mirrored in manufacturing philosophies like Integrated Product Development [1], Concurrent Engineering [2], and Product life Engineering [3]. The insight has also led to the development of many product development tools under the label DFX — Design for X, where X is either a product life phase or a "universal virtue".

Similarly to DFX's, Modular Engineering (ME) may be seen both as a philosophy / a structuring methodology and as a set of new tools that are related business operations. Especially ME is related to product development activities like market analysis and segmenting, conceptualisation in product design, choice of product architecture, structuring of production, and configuring sales products. ME may therefore be made an into a comprehensive company philosophy, an element for running the business. Therefore a metric for ME becomes a question of identifying and controlling the influence of ME on several areas, seen alone and in interaction, especially in the chain of interactions, recognised as the theory of dispositions [3].

2. The metrics of modularity

Literature proposes several methods for creation of modules and modular structures (where the concept module has various meanings and relates to many kinds of systemisation):

- Methods, where the versatility of the modular structure is in focus (i.e. how great is the number of possible variants that are created based upon few units). [3,4]

- Methods, where the module drivers are used as a starting point for modularisation [6]. The module drivers are special conditions, which in a particular situation indicate the need for encapsulation, leading to modules.

- Methods where the number of interfaces is being optimised [7]. In these methods one tries to minimise the couplings between units.

In the early 60's Borowski presented a standard for building block system [8]. The description of the system (Bauprogramm, Baumusterplan) was based on dividing systems to building blocks according to the range of use, types, elements, and assembly of specific types. In the development of building block system the point of view was both technical (size, tolerance), functional (requirements, range of use) and operational (combination, assembly, economy, arbitrary).

Diversity Cube was presented by Reinders [9]. It is a tool for diversity management during product development. Its' purpose is to quickly reveal the consequences of choices made in various disciplines. There are five kinds of choices, which are optimised by this tool. 1st Registration of the desired type of diversity within a product generation. 2nd Grouping the types together into commercial families. 3rd Grouping the types together into technical families. 4th Definition of modules or components, which is supposed to be done so that there are as few as possible different components used in the system (in each, as few versions as possible). 5th Concentrating of the remaining necessary diversity on as few components as possible.

Some matrix methods like Module Indication Matrix (MIM) [6], Design Structure Matrix (DSM) [7] have also been presented. The usability of these methods has been tested in industrial cases. Erixon has also introduced presentation tools for ME. These tools help to apply ME in developing product generations.

Three indices for modularisation were presented by Ishii [5]. These are Commonality, Differentiation, and Setup Cost Index. The indices measure the cost of providing variety, represent the variety, and measure the importance of variety. In the strategic level Garg & Tang [10] have presented postponement strategies for product families with multiple points of differentiation. There the main differentiation points are family differentiation point and product differentiation point. Ulrich has also presented some strategic methods and examples of statistical studies [11]. He has classified modularity into five categories and given optional strategies for modularisation. Both the classification and the strategies are elucidated with industrial examples.

2.1 Architecturing principles as the objectives of Modularisation

From studying literature and industrial cases we have got the insight that there are three properties that can be regarded the purposes or objectives for modularisation of a product architecture. We call these purposes the architecturing principles: Creation of *variety*, Reduction of *complexity*, and Maximisation of *kinship*. Thus, development of modular product architecture is based upon the architectural principles (see Fig 1). All the principles have strengths and they create distinct effects for product development, production, service, sales, etc.

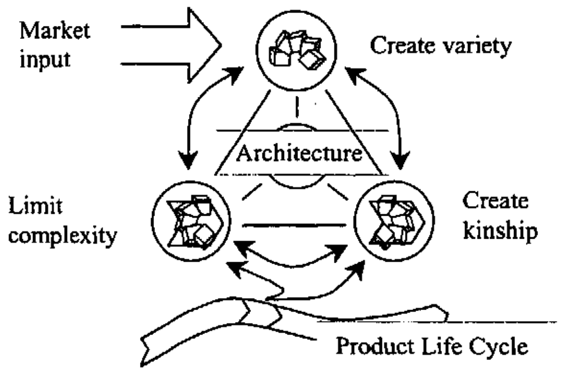


Figure 1. Three circumstances related to the architecture of the product family [12]

As the Figure 1 implies, the evaluation of the architecture is determined by three viewpoints, which seem to be independent from each other:

1. Creation of *variety* is necessary from a market and business viewpoint.
2. Reduction of *complexity* has an effect on making internal operations interdependent of each other and thus, to processes as a whole.
3. Maximisation of *kinship* is especially related to all product life cycle phases.

We will come back to the architecturing principles later in this paper.

2.2 Product quality versus modularity

Mørup has presented in his dissertation [13] two types of quality, which correspond to two types of quality stakeholders, the Q and q. The Q refers to the quality aspects seen from a point of view of the external stakeholders. The q is referring the quality aspects seen from the company (i.e. the internal stakeholders') point of view. External stakeholders are such as users, customers, dealers and other external sales organisations, approving authorities, and external service providers (independent from the initial production organisation). Internal stakeholders are all internal operational areas and employees in contact with the product (e.g. design, production, and quality control departments).

There are fundamental differences in the stakeholders' possibilities and ways to communicate the needs and expectations concerning the product. Internal stakeholders are often close to the design and production departments both in terms of physical and conceptual distance (i.e. they share cubicle and terminology). External stakeholders are remote; they have difficulties in expressing their needs, which sometimes are latent. Ultimately, they seldom complain but just change to a competing product or brand.

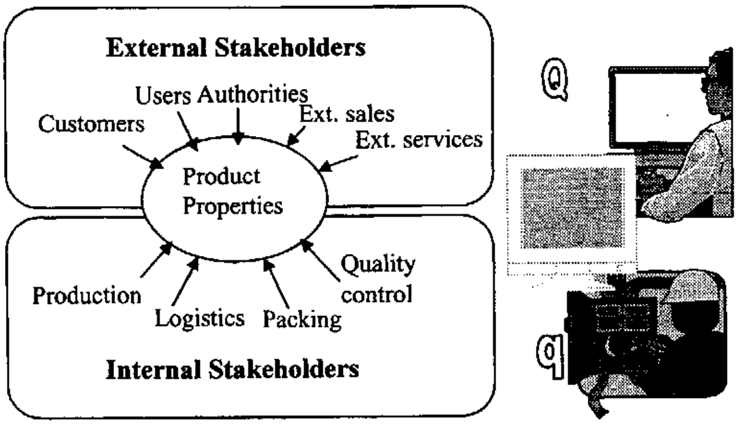


Figure2. The two product quality aspects — Q and q [13].

External and internal stakeholders look at different entities in a product. Within a producing company every entity is important and interesting because they create tasks to the organisation. In fact, people in a company have to consider entities from single components as process objects and cost carriers to the complete product as an assembly structure, inspection and testing object, etc. On the other hand customers look at the product as a whole, for instance functionality and image. Note that the customer can comprehend the whole with very polymorphic concepts, e.g. as "the quality of the details" or "good finish".

How different quality aspects (Q and q) are influenced by modularisation and ME? We may answer to this question with a statement that is seemingly a paradox:

ME does not directly influence the product quality,
but has an influence on handling of quality.

With systems approach to technical products [14] the superposition of structures (process, function, organ, and part) is presented. Modular structures are determined by the function structure, but the central issue is not how well does the module the function it is supposed to do. In module design e.g. the encapsulation (the definition of the ways that are allowed to be used in the interaction between modules) is a central issue. The module has to function and in this aspect in module design is not different from an engineering design in general. It is normally neither the intention nor purpose to affect the product quality through ME.

The encapsulation of modules leads to situations where quality problems are more easily (i.e. separately) managed:

- Modules allow Q-focus on module-oriented optimisation (reliability, robustness) in more stages.
- Modules allow q-focus on process and assembly optimisation (e.g. Taguchi method)
- Modules enable separate quality assurance according to decomposition.

On the other hand, experiences from DFA and DFM show that one have to be alert and not to lower the whole quality (Q) by redesigning products and parts for assembly or manufacturing. In several early DFA examples the original quality of a product was actually violated, when assembly features were optimised. In DFX-education at the Department of Control and Engineering Design (in DTU) we have been using the method "Know your Product" for ensuring that students have sufficient insight into functionality and quality of a product. Thus, the paradox should be carefully taken into consideration in ME.

3. Development stages and ME

In earlier sections of this paper, we have taken a look on the relation on metrics and analysis between modularisation and the object of it (i.e. the technical system that is being modularised). In this section we take a look on different methods of project management and their applicability in ME.

3.1 Development stages of ME systematics

By ME systematic, we understand the set of organisational efforts and their co-ordination in an enterprise as a basis for ME. Organisational efforts are the seven dimensions of the product development system, defined through the UNIC philosophy. The seven dimensions are organisational structure, decision structure, social system, methods and tools, knowledge structure, metrics and measuring system, and physical surroundings.

In a company, ME systematics evolve through a sequence of stages. We might not find clean-cut examples of enterprises, which have evolved through these steps. Instead of this we find some enterprises, where current state and organisation are in accordance to a one stage and where endeavours are set to reach another stage. Each of the stages (see the Table 1) has a network of involved internal functional areas and external stakeholders and thereby a set of corresponding metrics. Within each stage an enhanced instrumentation for the business activities in each operational area exists enabling improved co-ordination and integration.

We have chosen to use the set of five development stages. Note the synchronous growth of the range of organisational structures and the importance of modularisation through the stages.

1. Manufacturing of independent products

Product Planning & Development	Independent development of each product, which obtains its completing edge based upon its own functionality and qualities.
Products	No structured kinship between products.
Production	No structures concern from design to production (DFMA).
Product life aspects	Product life aspects not utilised for competition.
Knowledge management aspects	No organised or structured knowledge handling.

2. Product families as a market strategy

Product Planning & Development	Co-ordinated development of products into product families.

Products, Production, Product life aspects, and Knowledge management aspects as in stage 1.

3. Architecture related manufacturing

Product Planning & Development as in stage 2.

Products	Architectural commonality of a product family's products.
Production	Production planned for unit or module production.
Product life aspects	To a certain degree, product life aspects are utilised for competition.
Knowledge management aspects	To a certain degree, knowledge management in accordance to product architecture.

4. Configuration oriented manufacturing

Product Planning & Development	Controlled and managed based upon family architectures, platform definitions and market needs.

Products are as in stage 3.

Production	Unit oriented (concerns also suppliers).
Product life aspects	Product Development is product life oriented.
Knowledge management aspects	Product Data Management and Knowledge Engineering structured like the product family architectures.

5. Dynamic Modularisation approach

Product Planning & Development	New product development based upon creating or purchasing new units or modules, in accordance to dynamic market demands and platforms.
Products	Flexible architectures accepting new modules and deleting old ones.
Production	Units / modules from a mass customisation philosophy (also at suppliers).
Product life aspects	Product life fit and qualities utilised for competition.
Knowledge management aspects	Dynamic innovation of module and architecture related knowledge. Knowledge Empowerment as a strategic element.

Seen in relation to ME metrics, it is obvious that it is not possible to use the same metrics in all stages. This is due to measured issues: the organisation and the business objectives. In different stages of ME the organisational units are different in span and responsibility and business-oriented goals are different in each stage.

3.2 Modularisation stage

Modularisation of a product family is a central aspect in ME. In a distinct enterprise, where product family issues have been confronted, the question may be raised: how far has ME evolved? On what stage is modularisation? We see that instead of questioning "in which development stage of ME systematics a company is", the architecture of a product family has to be evaluated. This is inevitable, because companies usually have different product lines that represent different development stages in ME systematics. Thus, we raise the questions: Can we make graduation for architecture? Can we speak about more or less well-carried modularisation? Are we sure that certain architecture with a certain set of characteristics is an optimal one?

The evaluation of the product family architecture should show good results in every three architecturing principle (see figure 1 in section 2.1). Despite this, the caused positive or negative effect in each principle is going to be very dependent of the enterprise: the actual business context (i.e. the business processes, organisations, and production conditions).

Each principal form of product architectures (modular, integral, combinations and mixed systems, parametric, etc.) has its' own rules for how the structure will be, i.e. if we are developing functional or non-functional units, standardised interfaces, etc. Therefore it seems to be impossible to create universal metrics, related to modularisation stage, for evaluating the excellence of product family architectures. Instead of this, one has to look in each case how well the current modular system meet its' life cycle requirements, what are stakeholders opinions about the architecture, and how to manage the ME projects.

4. Evaluation of the architecture

4.1 Life cycle of the modular system (LCMS)

The Life Cycle consideration is very important in ME systematics, because the investments must be covered in the long rung. We suggest a checklist approach for evaluating the ME systematic.

	Weight	Score
1. Has the life span planning for the modular system been made?	#	#
2. Has the core product been divided into modules based on the periods certain technologies are being used?	#	#
3. Can modules, which have shorter life span than the product, easily be replaced with new ones?	#	#
4. Does subcontracted key modules fit to the life span of the system?	#	#
5. Does a frame make it possible to change the modules (i.e. update the product)?	#	#
6. Can the appearance of the product be reformed, renewed, or modernised?	#	#
7. Can the environmental consciousness be improved by changing modules?	#	#
8. Has the Modularisation Implementation Profile been analysed?	#	#
9. Is the lower level modularisation made by using similar principles as the core product?	#	#
10. Has the responsibility of the maintenance of the dynamics of the modular system been appointed to a certain person or a team?	#	#
Summary		#

The characteristics of evaluated LCMS is	#
In a superior system it usually is	#... #
In an average system it usually is	#... #
In a poor system it usually is below	#

4.2 Stakeholder analysis

There is an analogy between stakeholders in Design for Quality and stakeholders in ME. For an internal stakeholder, ME is a promising means towards enhanced productivity, but it has also an affect on business circumstances, e.g. in handling the supply chains and relocating production. The prediction of module availability will become more difficult when totally unlike companies with dissimilar products apply the same modules. This can mean that sometimes the shortage of modules occurs even when the market research of a company does not indicate it. Production relocation has its' basis in global outsourcing thinking.

On the other hand an external stakeholder (e.g. customer) can expect increased superiority from his/her viewpoint. ME enables mass customisation and late point differentiation. This means that the individual customer may have an individual product in a short time, but also here the business circumstances are changing to direction that is not always better than earlier. Some companies, which have been able to reduce delivery time a lot, are delivering products only for customer orders. However, to a customer the delivery seems to take longer than earlier. For instance, many car manufacturing companies produce cars only for customer orders (i.e. not to stock as earlier). Earlier customer could get car right away from the dealer, but nowadays he/she has to wait up to five weeks to the individual car. The total manufacturing process is faster and more inexpensive (due to reduced warehousing), but not from the customer point of view.

If we look the situation from the viewpoint of another external stakeholder, the society, there is also a transition going on. Earlier the big companies were very stable employers. Nowadays the old facilities are more an obstacle for the dynamically changing company than something that creates competitive edge. Increased concentration of knowledge is becoming one of the most important reasons to keep a company in a certain area.

Even these simple examples show that at the same time ME can have both positive and negative influence to different stakeholders. To understand better these consequences we have defined three key characteristics, which have been used as classifiers for each stakeholder.

Three of the most important characteristics in the company's point of view are:

1. Profitability
2. Growth
3. Knowledge potential

All the three characteristics have to be superior to competing companies. When approaching the more general characterisation to cover also external stakeholders, our conclusion was to use three characteristics that have to be itemised with some stakeholder characteristics:

1.	Awareness	Profitability
2.	Capacity	Growth
3.	Capability	Knowledge potential

While other characteristics are used in their contemporary meaning, the term Knowledge potential has been created to describe widely needed knowledge, its' cultivation, institutional learning, and maintaining its' consistency. Knowledge potential is created in a situation where there is a concentration of knowledge on a certain discipline or topic, in a certain (physical) area that it creates a potential (analogous to potential energy) that enables the technological (or other) development in that area. To created needed knowledge potential the topics are at least technologies, customer groups and applications, but above all key views of all stakeholders for the future, as well as trends for scenarios and forthcoming technologies are needed.

In Table 1 we present some of the general observations for how ME systematics can have an influence on stakeholders. In the table the above-presented characteristics function as a framework. This presentation is supposed to serve as an inspiration to make a more detailed stakeholder study (i.e. stakeholder analysis). When ME systematics is considered, the first task is to create a number of product concepts. The strength that lies in generating a number of concepts is proven by the Second Toyota Paradox [15]. The second task is to perform a Stakeholder Analysis, which gives better insight for selection of concepts that are further developed. Using the concepts presented here, gives a better understanding for the goal setting of a ME systematics. We suggest that some company specific weighting (based on the company's strategy) should be used for getting better results.

Table 1. The charts for making stakeholder analysis of modularisation effects

Company metrics	**Customer / Distribution**	**End Customer**
Profitability Complexity will be lowered. Product management is possible. Cost reduction.	Awareness Complexity will be lowered.	Awareness Complexity will be lowered.
Growth Needed capacity is smaller in the operation. Localisation is easier through modules.	Capacity Localisation is easier.	Capacity Lead-time is shorter, flexibility is gained.
Knowledge potential Company's capacity will be directed to increase and maintain core and integration knowledge.	Capability Capacity can be directed to maintain the application knowledge.	Capability Risk, Environmental effects.

Table 2. The charts for making stakeholder analysis of modularisation effects

Environment	**Company management**	**Company staff**
Profitability Possibilities for environmentally conscious operation will be increased.	Awareness Right investment policy for modularisation is needed. Management of effectiveness will change from operation to planning.	Awareness Dynamic situation, only profitable business gives workplaces. Everybody will own responsibility of a profit making in a company.
Growth Society must make it attractive to work in module producers. Localisation through modules becomes easier.	Capacity Capacity management in own company becomes easier. New co-operation models are needed to manage the whole supply chain.	Capacity Pressure on diminished lead-time and increased flexibility. Capacity will be directed to more knowledge intensive areas.
Knowledge potential Model of the effects of globalisation is needed. Knowledge intensiveness increases.	Capability The management of knowledge is increasingly difficult because of information overflow.	Capability Worldwide competition on the issue: who has the knowledge that is approved and used.

5. Management of a ME project

5.1 Modularisation Implementation Profile, MIP

The creation of a Modular System needs plenty of resources and it is necessary to evaluate how well the modularisation process will be performed in a company. For this evaluation we suggest using a Modularisation Implementation Profile, MIP. MIP is another method based on giving grades to certain common questions that are related to modularisation process. Here is an example of the most common questions:

1. Modular System Mission	#
-Are the goals clear and understood by all significant stakeholders?	#
-Does missions cover the whole life cycle of modular system?	#
Sum 1	#

2. Top Management Support	#
- Does top management allocate sufficient resources?	#
- Does top management support in the event of crisis?	#
- Does top management support rewarding the final result?	#
Sum 2	#

3. Modularisation Plan	#
-Does the plan cover work breakdown structures with important milestones, resource assessment, and project-monitoring mechanisms?	#
-Is the detailing (not too detailed) of the schedules, milestones, manpower, and prototype requirements sufficient?	#
-Is the plan capable to utilise the performance of the suppliers?	#
Sum 3	#

4. Client Consultation	#
-Are the clients identified and their needs clarified?	#
-Are clients' needs easy to utilise for supporting decision making during the execution of the development?	#
-Are scenarios for future needs of clients created?	#
Sum 4	#

5. Resources	#
-Has the development project enough (technically) skilled personnel?	#
-Has the development project enough (administratively) skilled personnel?	#
-Is the development project team dedicated to the goals to be reached?	#
Sum 5	#

6. Technical Task	#
-Is the product development available sufficient for to the reach the goals?	#
-Is IT-implementation (PDM, design support and MRP systems, etc.) available to the reaching the goals?	#
-Are the technologies used valid for the whole life cycle of the product?	#
-Have enough number of alternatives been studied?	#
-Are tolerances supporting the dynamics of a set-based design?	#
Sum 6	#

7. Client Acceptance	#
-Is the client acceptance for the Modular System products been ensured?	#
-Is a verification of client acceptance performed with the use of prototypes or other types of visualisations (e.g. with virtual reality applications)?	#
-Are the product scenarios verified with clients?	#
Sum 7	#

8. Monitoring and Feedback	#
-How is the development comparing to its initial projections?	#
-How is the performance of the development team?	#
Sum 8	#

9. Communication	#
-Are design sets known in the team and in the organisation?	#
-Is the information exchanging at milestones organised perfectly?	#
-Are constraints made known in the organisation?	#
Sum 9	#

10. Troubleshooting	#
-Is the organisation capable to find real sources of problems?	#
-Is the organisation powerful to solve problems?	#
-Are the problems transferred to the "lessons learnt box"?	#
Sum 10	#

To perform the above checklist the expertise is needed. In the future our aim is to collect a statistical data to support the study.

5.2 Decision making in the Modular System development projects

We can see from what we have above stated that it is a versatile decision making process related to an establishment of a modular product architecture. As presented in 3.1 we have chosen the names and main characteristics for the five stages of the development procedure. Between stages there are transferring projects that change both the business units and products from one stage to another stage. The nature of decisions in managing transferring projects are very similar to selecting which projects can be realised in New Product Development. Cooper has presented a Stage Gate method for New Product Development. In it, the decision gates have format as follows (see also the Fig. 3):

1. Inputs: A prescribed list of deliverables that is supposed to present at the Gate Review.
2. Criteria: A set of hurdles, criteria, or questions that the project is judged on.

3. Outputs: A decision: Go/ Kill/ Hold/ Recycle an action plan or path forward.

We are going to apply similar systematics. Each Modular System development project is evaluated whether it can be continued, stopped or fixed before significant resource is used on them.

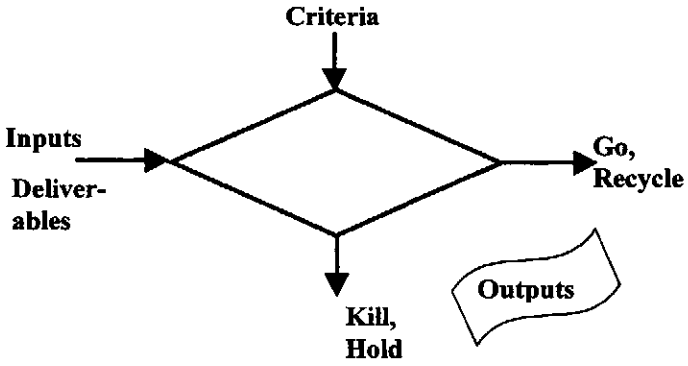


Figure 1. The structure of the gate.

Gates deal with three critical issues:

1. Is this the right project on which to spend company's valuable and scarce resources on? Is the project a quality project with value to the company that justifies further expenditures? Is it a good business proposition?

2. Is the execution of the project going to be handled in a quality fashion? Are the project leader and the team members doing their job in a quality fashion? Are the deliverables to the gate in good order?

3. Is the "path forward" a quality one? That is, is the proposed action plan reasonable, and are the resources proposed — people, money, and time — appropriate and realistic.

In every gate the format is common to each other. The elements of gate (deliverables, criteria, and outputs) have a certain role that enables a combination of quality projects in a company.

Deliverables are what the project leader and team members must deliver to the gate. They are the results of actions in the preceding stage and thus they form the objectives of the project leader and the team. Standard lists of deliverables are defined for each gate. As well, at the preceding gate, both the path forward and the deliverables for the next gate are decided. *Criteria* are what the project is judged against in order to make the Go/Kill and priority related decisions. These criteria are usually a standard list for each gate, but usually change from gate to another. They include both financial and qualitative criteria, and are broken down into required characteristics versus desired characteristics. *Outputs* are the results of a gate and they include the result of a decision

(Go/ Kill/ Hold/ Recycle) and a path forward (an approved action of project plan and the list of deliverables for the next gate).
The essential problem in project evaluation at the earlier gates is allocating scarce resources across different projects, each at different stages of completion. In practice, the gate decision breaks down into a two-part decision.

The 1st decisions are related to the question: is this project a good one? If this were the only project available, would we proceed with it? Here the project is evaluated against its own merits against a set of standards. This decision is a "Pass vs. Kill" decision.

The 2nd decisions are related to giving priorities. Given that the project is a good one and considering the other projects are already underway and thinking the resources available to all of the projects, what is the priority of this project among other projects? Is it a one with a high priority — a strong green light and emphatic Go — or is it a Hold–one that we put on the shelf until resources become available at a later date?

As an example we present a gate from *Architecture related manufacturing* to *Configuration-oriented manufacturing.*

Gate 3:	From Architecture related manufacturing to Configuration-oriented manufacturing.
Objective:	To make commitment to full Modular System development.
Method:	Tolerances for design sets, prototypes; Modularisation Implementation Profile (MIP); Life Cycle of the Modular System (LCMS); financial review based on Net Present Value and Internal Rate of Return together with sensitivity analysis, a portfolio impact assessment via portfolio maps.
Gatekeepers:	Key Functional decision-makers in the business unit.

6. Conclusion

In this paper the concept of systematics for measuring modularity. The concept has been developed in Technical University of Denmark and it is still under development.

A good modular lowers complexity, gives markets the needed variants, and supports the whole Life Cycle of the product through creating kinship between parts and components. Based on this definition, we suggest measuring system for the evaluation of modularisation. It has the following iterative steps: measuring the modularisation state of business in a company, performing weighted stakeholder analysis, establishment of a screening system for supporting decision making for modularisation, and normal economical calculations, two checklists analyses: Life Cycle of the Modular System (LCMS) and Modularisation Implementation Profile (MIP).

Checklists are used widely in the measuring systematics, because they are the simplest and perhaps the more immediate aids to design deliberation and similarly to modularisation. The presented checklists are prepared on the assumption that requirements, which have been overlooked earlier, will be overlooked again. So the useful data for making comparison between incremental steps in modularisation is collected.

Our aim is to develop the system forward to a more detailed level, verify it with industrial examples, and finally develop computer support prototypes to be used in performing the measurement (i.e. collecting the metrics data and making the comparison).

Metrics for modularisation and metrics for product development have similarities. In the future modularisation is a part of product development as it is now in Toyota's Set-based Engineering. The interfaces of the modularisation and the product development processes and their metrics have to be defined.

Acknowledgement

The authors express their gratitude to the Academy of Finland and Technical University of Denmark for supporting the research presented here.

References

[1] Andreasen, M.M., Hein, L. *Integrated Product Development*. IFS (Publications) Ltd. Kempston, Bedford. Uk. Springer-Verlag. 1987.

[2] Winner, R. I., Pennell, J. P., Bertrand H. E., Slusarezuk, M. M. G. *The Role of Concurrent Engineering in Weapons System Acquisition*, IDA Report R-338, Institute for Defense Analyses, Alexandria VA. 1988.

[3] Olesen, J., Wenzel, H., Hein, L., Andreasen, M.M. *Design for Environment* (in Danish), Miljø- og Energiministeriet 1996, ISBN 87-7810-435-1.

[4] Janson, L. *Activity – fitted Product Structures* (in Swedish), Ph.D.-thesis, Department of Production Engineering, Chalmers Technical University, 1993.

[5] Ishii, K., Juengel, C., Eubanks, C.F., Design For Product Variety: Key to Product Line Structuring. In: *Proceedings of the 1995 Design Engineering Technical Conferences De-vol. 83*. Vol. 2 ASME. 1995

[6] Erixon, G. *Make the Products Modular* (in Swedish), Konstruktionsdag 1997, Technical University of Denmark, Lyngby 1997.

[7] Pimmler, T.U., Eppinger, S.D., Integration analysis of product decompositions. In: *DE-Vol. 68, Design Theory and Methodology — DTM '94*, ASME 1994.

[8] Borowski, K-H. *Das Baukastensystem in der Technik*. Springer -Verlag, Berlin, Göttingen, Heidelberg. 1961.

[9] Reinders, H. *A Tool for Diversity Management during Product-Development*. CFT-technology. Report 24/89EN, Philips Centre for Manufacturing Technology, 1998

[10] Garg, A., Tang, C.-S. On postponement strategies for product families with multiple points of differentiation. In: *IIE Transactions (1997) 29*, pp. 641-650

[11] Ulrich, K.T., Ellison, K.J. *Customer Requirements and the Design-Select Decision.* Working Paper. Department of Operations and Information Management. The Wharton School. 1997

[12] Riitahuhta, A., Andreasen, M.M. *A theory of Structuring of Product Families.* Department of Control and Engineering Design. Technical University of Denmark. (Planned)

[13] Mørup, M. *Design for Quality*, Diss. Institute for Engineering Design, Technical University of Denmark, 1993.

[14] Andreasen, M.M., Hansen, C.T., Mortensen, N.H. The Structuring of Products and Product Programmes. In: *Proceedings of the 2nd WKD Workshop on Product Structuring 1996.* Delft University of Technology.

[15] Ward, A., Liker, J.K., Cristiano, J.J., Sobek, D.K. II, The Second Toyota Paradox: How Delaying Decisions Can Make Better Cars Faster. In: *Sloan Management Review / Spring 1995.*

A. Riitahuhta
Tampere University of Technology
Machine Design Laboratory
Korkeakoulunkatu 2, P.O. Box 589
FIN-33101 Tampere
Finland
tel. +358-3-365 2241
fax. +358-3-365 2307
aor@ruuvi.me.tut.fi