INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 03 STOCKHOLM, AUGUST 19-21, 2003

DYNAMIC MODULARISATION - A CHALLENGE FOR DESIGN PROCESS AND PRODUCT ARCHITECTURE

Timo Lehtonen, Tero Juuti, Antti Pulkkinen, Asko Riitahuhta

Abstract

Currently there exists a need to produce product variants in faster and faster pace due to turbulent markets. Modular product structures are regarded as suitable for this. However committing to the modular product family is normally considered as a commitment to a certain architecture, which is often seen inflexible when compared to one-off-product development. Dynamic Modularisation (Dymo) business and product development paradigm aims to maintain the agility of one-off-product development when producing modular product families [1,2]. Now Dymo is developing from a proposed paradigm to an industrial practice. At this stage there is now experiences available, what kind of challenges a company is about to face when it adapts the Dymo type way of working. In this paper we present the fundamental background behind Dymo and an example of the implementation of Dymo. We discuss challenges in implementation and propose possible solutions.

Keywords:Design management, Business process re-engineering, Platform design and Modularisation - product families

1. Introduction: Modularity as a solution to variant explosion

There are many companies, which have noticed the traditional approaches i.e. mass production and project-based deliveries are insufficient for making profitable business. The business results in numerous branches of industry illustrate the underlying deficiencies of traditional paradigms.

In mass production the problems are due to the increased number of product variations needed in meeting customer expectations and in the project business the fierce tempo of competitors introducing new products. The amount of variations has increased exponentially, which consumes considerable amount of engineering effort that is needed in designing all variants and maintaining the documentation. The huge number of variants also cuts down the manufacturing batch sizes and economies of scales with the expense of cost efficiency.

Several approaches have been introduced to solve this problem [3,4]. The companies applying these methods have been able to manufacture customer variations cost effectively alike mass production previously. Their success has propagated pressure on project business also; Cost effectiveness and the urge for operative efficiency have funneled these companies towards mass customization.

Product structuring is the major but quite often neglected resource when adapting mass customization. Nevertheless the modular product structure provides the most promising solution for the industry with mechatronic products. The modular structure can be achieved in various manners depending on the definitions of module, modularity type and viewpoints in the company. The modularity *per se* is not a virtue nor it solves anything just by its being. This is to say that the companies also need an effective operational mode to benefit from common design base resulting sufficient diversity in the product offering.

2. Different definitions of modularity

In research community the concept modularity has been defined with different ways. For example Ulrich defines modularity as a property of product structure and defines 5 types of modular structures (sharing, swapping, cut-to-fit, bus and sectional modularity) [5]. In American research modules are not always defined at all but instead a word "chunck" is used to describe loosely defined part or subassembly of a product for example [6].

In his Ph.D thesis Erixon lists 12 reasons or motivations, which he calls Module Drivers, to form a module. These are Carry over, Technology push, Planned change, Variance, Styling, Common unit, Process/Organisation, Separate testing, Black box, Service/maintenance, Upgrading and Recycling [7]. If we take more abstract view on these motivations, we found six groups of motivations; Component-sharing and re-use, long term product management, life-time maintenance of individual delivered products, organisational production oriented reasons, recycling and product variance.

The actual module itself is quite seldom defined. In DFC-project [8] a module was defined by following axioms. This more focused definition of module can be referred as m-modularity.

Part or sub-assembly of a product is a module when
1. It has defined interface, which determines its connection to other modules.
2. It is a member of set of parts/subassemblies, which forms a modular system.

In this definition, system is considered as a modular system, when there exists some or all of those structural variations, which were defined by Ullrich to be types of modularity.

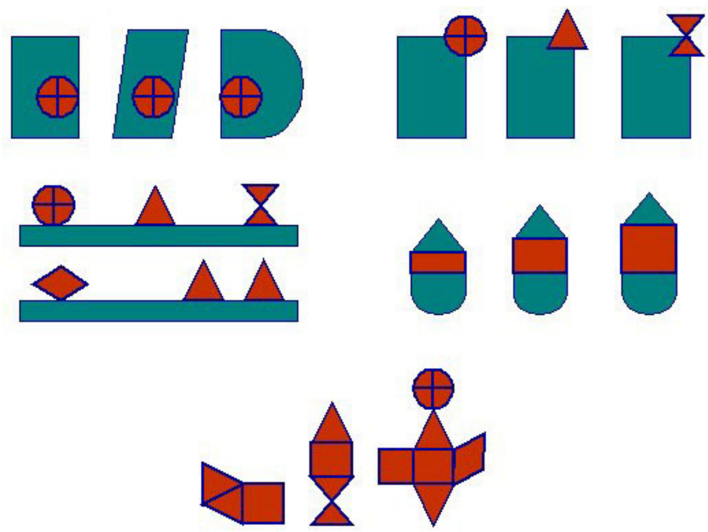

Figure 1. Ullrich types of product modularity; componet sharing modularity, component swapping modularity, cut-to-fit modularity, bus modularity and sectional modularity [5].

In addition to previous axioms, there is a requirement that module system must be defined on

one level only in concrete presentation (eg. part-domain). In other words this means that no module should include or consist of other modules of same module system. If this is allowed, the interfaces of modules could no longer be defined unambiguously and modules could no longer be handled as undividable black boxes.
In this definition it was not purposely defined according how the modules should be composed. The widely accepted idea is that modules should be composed of according the functional structure of the product. However, for example majority of Erixon's module drivers does not respond this. Besides, many successful modular products are divided according to assembly structure rather than functional structure.

In this paper we focus on customer tailored products and in this case the preference is clear. The core of the paradigm is answering to customer requirements in rational and systematic way. Thus the handling of requirements is the key point and normally the requirements are more or less related with functions. The assembly based modular structure could be used only, when product assembly structure resembles the functional structure (or functional requirements from markets/ customers). So whether it is difficult to achieve or not, mass-customised products normally require function based modular structure.

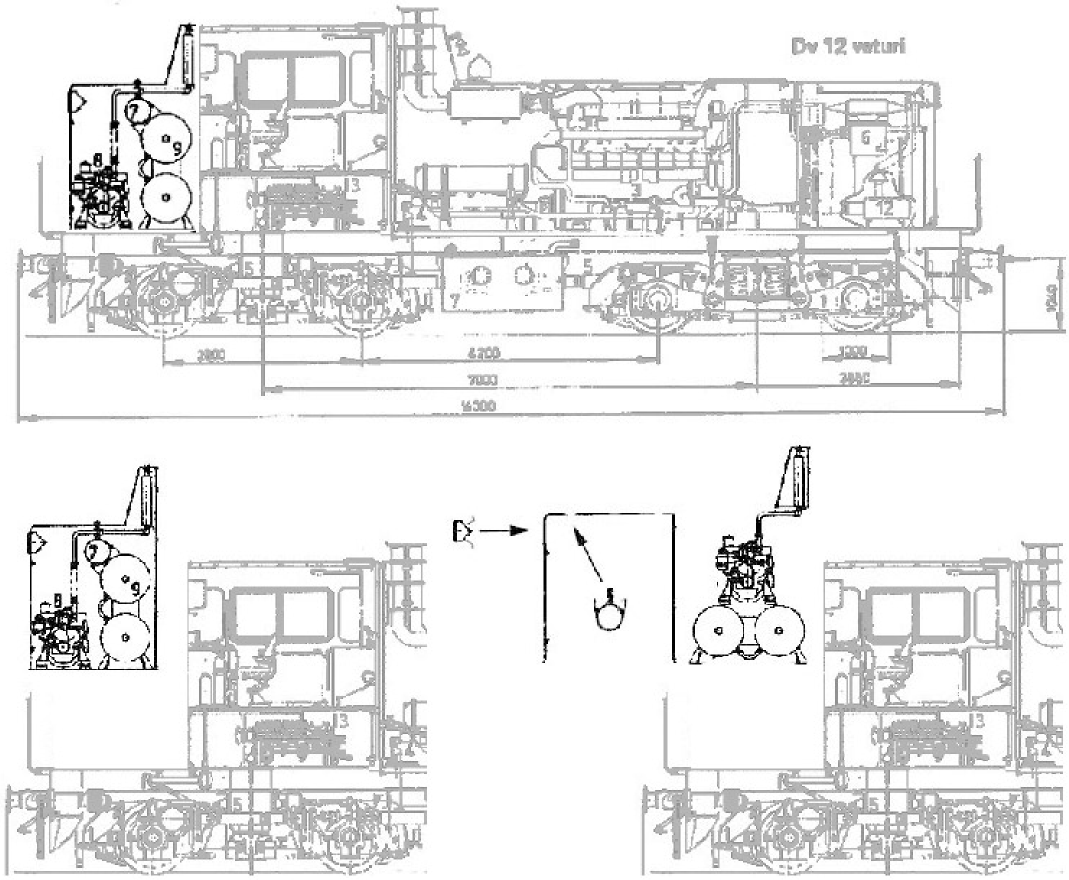

Figure 2. Locomotive production development in Transtech Tampere from sixties to nineties. Picture on the top shows "ad-hoc" integral structure of a Dv12-locomotive with integrated compressor cabinet marked with black. Picture below it on the left side shows assembly oriented modular structure where compressor cabinet has a separate frame. The picture on right shows effects of function based modularity, where compressor unit is one module and parts belonging to other functional modules (headlight, upper fuel tank and covers) are separated. For clarity sake, all pictures are drawn according a locomotive model drawn in sixties, although this particular model was never made in modular form.

3. Development of modular product structures

At late stages of the Second World War the German shipyards re-arranged their submarine production. They started to build the hull of a submarine of lengthways blocks, which they called “modules”. These could be considered for assembly based modules. These kinds of modules have been since largely used, but as noted above, this kind of structure is not exceptionally beneficial for mass customising.

Next logical step was function-based modularity. This type of structure can be derived for example from Theory of Technical Systems [9]. Also there exists abstraction of designed artefact, which supports this kind of structure. For example Domain Theory [10] presents that there exist an Organ-domain between the Function-domain and The Part-domain. This approach supports designing functional assemblies, which in modular product form the modules.

In mass-customisation a matter of great importance is to have control on variety. The function-based modularity does not address whether modules are standard modules to whole product family or are they (customer) variant modules. The state-of-the-art solution is platform-based modular structure, where the product is divided in standard sections and customer variable sections. The variable sections should have function-based modular structure, but the standard section could have assembly-based modular or even integral structure. It is important to note that only a fragment of product platforms used in industry is based on functional modules. There are other possibilities to form a platform as shown in figure 3. The platforms could even be set-based without any utilisation of modularity.

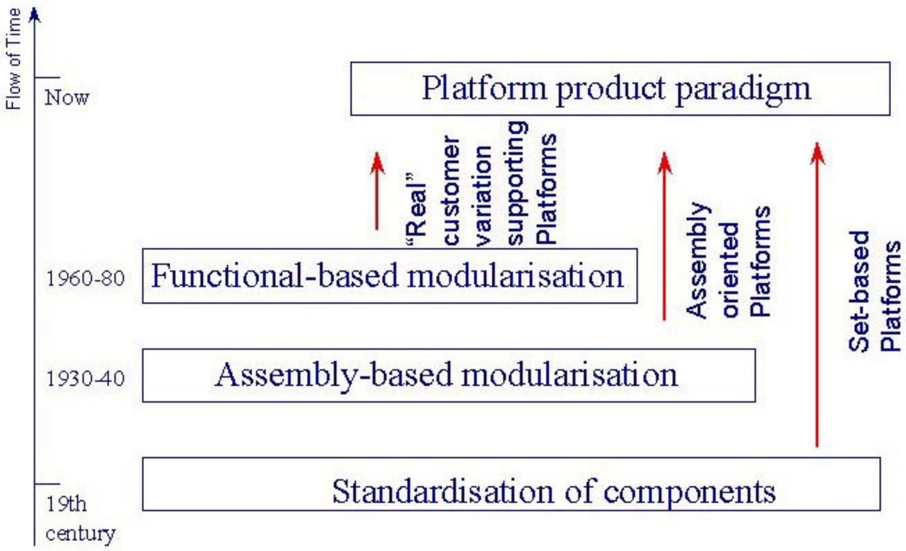

Figure 3: Different approaches to form product platforms [11].

If we claim that platform-based modular structure could cope with customer variation in mass-customising paradigm, there still exists one challenge: the variation within the product family life-cycle. We have earlier presented Dynamic Modularisation (Dymo) business and product development paradigm, which adds the platform-paradigm the company processes that are needed in handling the life-cycle variation [1,2]. The core idea of Dymo is making product development work on two levels. On the upper level there is platform development. This includes the customer requirement management, product architecture management and development and module creation process, where suitable modules are developed for fulfilling customer requirements. All these actions are targeted for creating a product platform, which enables launching a product family, which corresponds to market needs now and in predictable future.

The actual product creation happens in the lower process level. We don't use the word product development here, because the work is more a kind of integration of the modules than developing something completely new. This way of working aims to maximal design re-use. The business goals are shortening the development time for single products and increasing the productivity in product development.

The development of deploying modular product structures in industry can be seen as an evolution. This is presented in figure 4 below. At the bottom of the figure there is assembly based modularity. The next step above it, is function based modularity, which suits better to customer variant products than pure assembly based modularisation. Next step ahead is customer oriented platform based modularity, where customer variant and non-variant part of the product are separated. At the top of the picture we see Dynamic modularisation, where also the product family life-cycle is taken in account. This can no longer be achieved only by means of product structuring, but company processes as shown in figure 5 are crucial part of it.

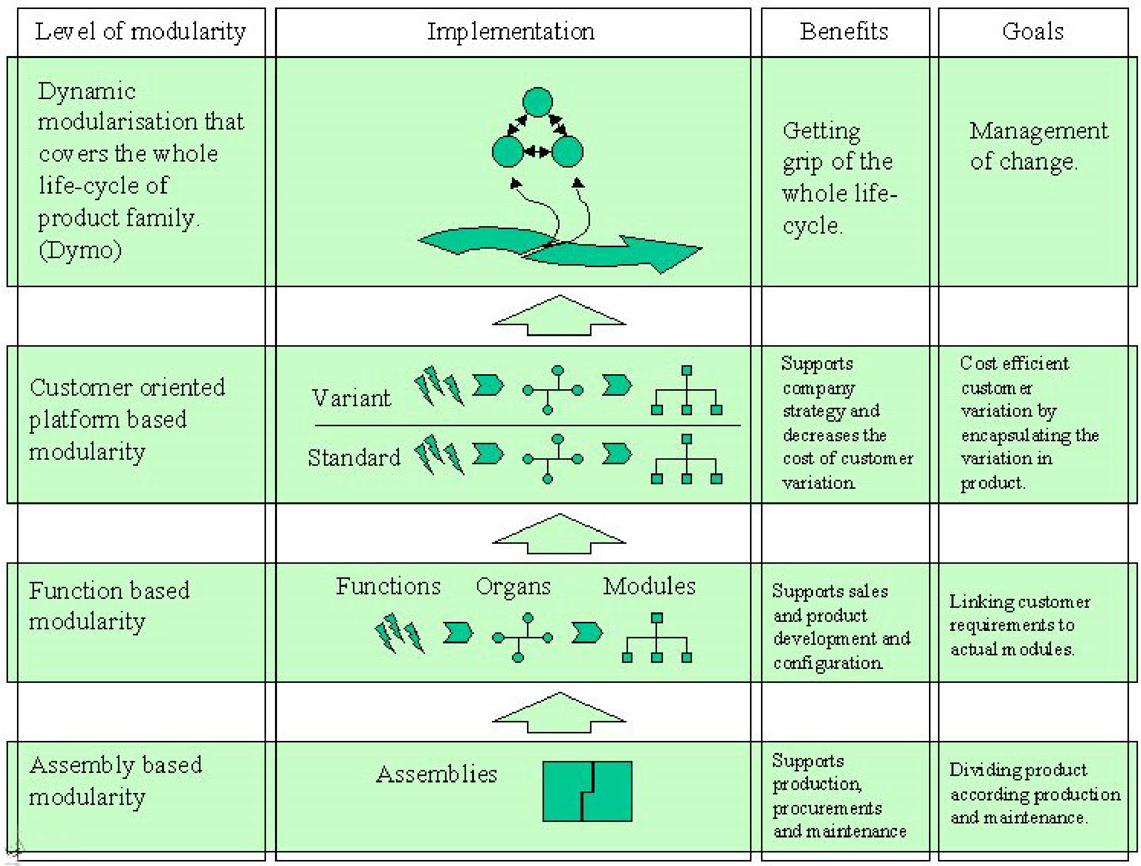

Figure 4: The development of deployment of modular product structures in industry represented as an evolution. This classification is not often easy to observe in real cases, due the fact that in reality the way of working is often a mixture of elements from different levels.

4. The implementation of Dynamic Modularisation

The Product Creation process is built upon using platforms and dynamic modularization. The process starts from defining *business needs* in various forms such as product categories, feature roadmaps, product roadmaps or product portfolios. The product level system architecture dictates interfaces and modules, which can be used.

The system level module structure is evaluated to establish platform capabilities. Features, functions of the product, product cost etc. are subset of capabilities. Then subsystems and their module structures are defined accordingly. Usually an organization is responsible for developing and delivering releases from their platform.

The releases are provided to product programs, which integrate them into final product. The product program selects suitable configuration of releases to meet their customer expectations.

Practical implementation of Dymo-process is shown in figure 5. Starting from the left we see how product architecture is formed according the business needs. The available platform has capabilities which are constrained by system level architecture. Subsystems are derived from the system level architecture. Modules/Componenets are released for the product program according to the subsystem architecture. Product program integrates ready made modules into product release. For this reason product programs are not called “development” but “integration”.

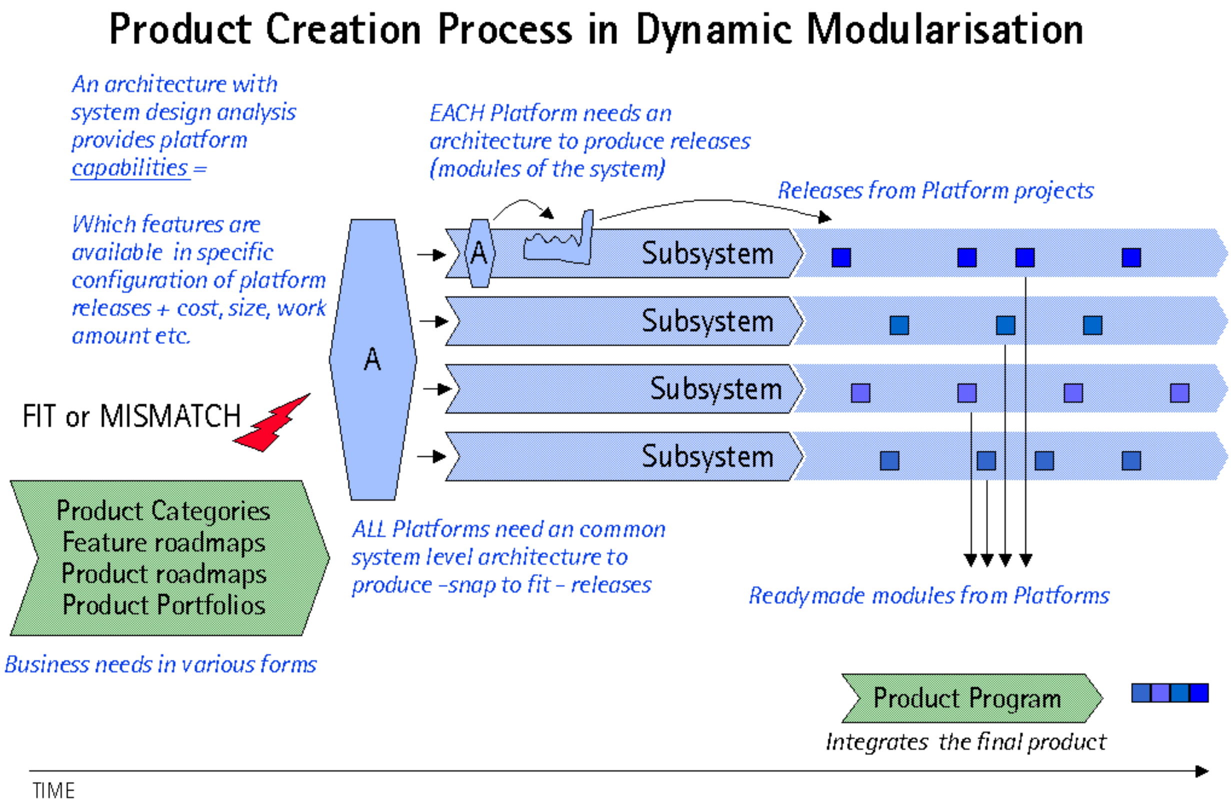

Figure 5. The implementation of Dymo-process in practice.

The process seems static, but there are several change sources for system level and subsystem level module structures. Main pressure for changes can be allocated to the module properties and to its behaviour. Key issues is the lifecycle of the module; when is it available and for how long. New technology as a strategic or competitive reason also imposes changes to the module and occasionally to the module structure, too.

In figure 6 we see the cause-effect chains in product structure evolution when utilising Dymo. The ovals marked with dark colour are vital to Dymo way of working and need not to be considered in one-of product development. The cumulative effect of increased interactions increases considerably interations in the product structure development.

The module structure needs to be modified also if the intelligence of the module is changed or partitioned into other modules. If the master module of master-slave bus is without processing power (i.e. brains) the bus is ineffective. The complexity in managing dynamic module structure is easy to understand if bearing in mind that several products, tens of products are affected if the module under change is planned to be reused in those products. Another critical issue with dynamic structure is time. Again several product programs are affected if the module project is delayed.

The major difference to platform based product development is the dynamic module structure. Currently the understanding is that the platform module structure needs to be somewhat static to be manageable and beneficial for the company.

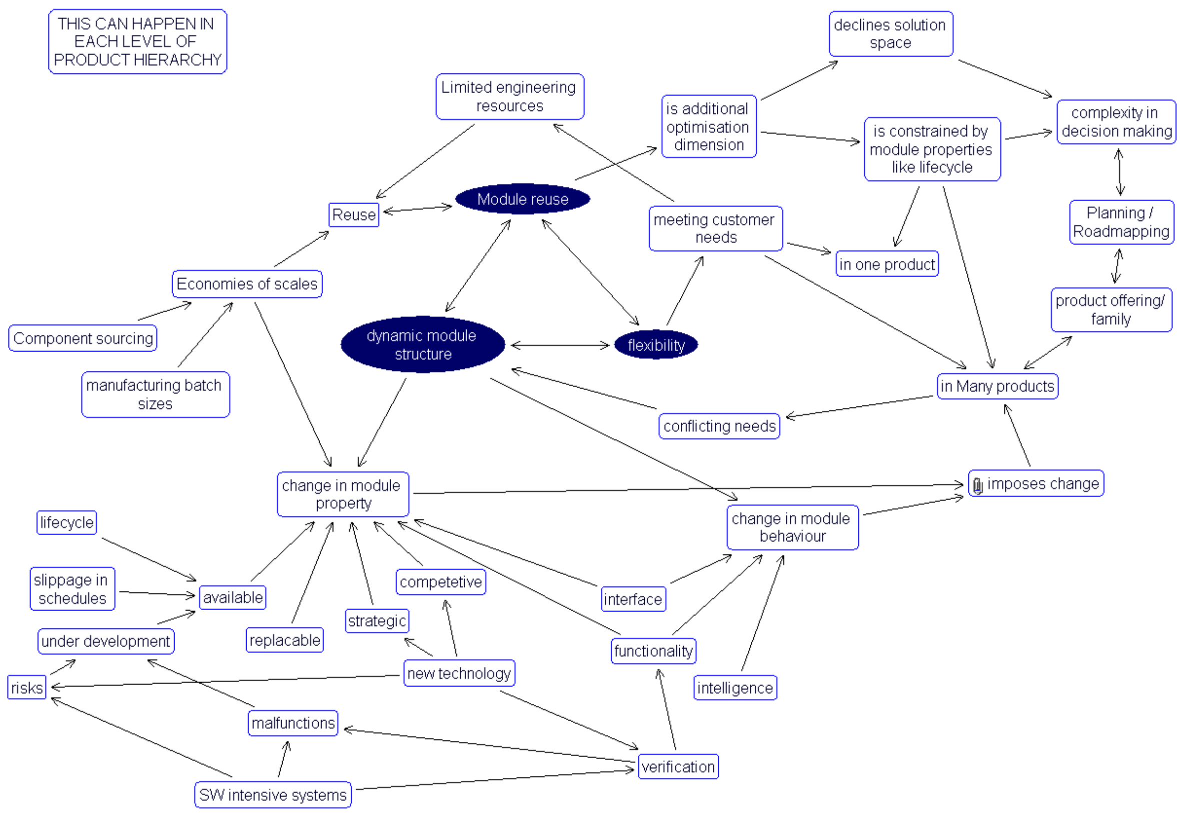

Figure 6. Cause-effect chains in product structure evolution when utilising Dymo. The ovals marked with dark colour are special to Dymo way of working.

5. Challenges in implementing Dymo

Experiences show that three types of challenges would be faced in utilising Dymo-paradigm: the challenges for processes, competence of personnel (mainly designers) and product architecture.

5.1.Challenges for processes

The scope of requirements will be enlarged and the requirement management will be partially dispersed inside the company. Also the whole nature of requirement management will change, as earlier it has been a part of every product development project, but now it becomes a constant process running all the time. The business unit needs to take into consideration more aspects than in one-of-product development. This is due the fact that requirements are no longer set to single products, but to a module system that is used in many different products. Also, the best possible knowledge about market situation and trends is needed. Thus, it seems probable that Dymo is suitable paradigm for market leaders working in global environment, but may be unachievable for example small local companies.

The other challenge for processes is the scattering of the requirement management. The module development projects have their requirements from business unit, but their internal customers are the product programs. Naturally business unit also controls the product programmes, but evidently very effective and reliable communications are needed. This underline the urgent need of common terminology and understanding inside the company.

During product development verification and defect management processes are needed, too. The dynamism in the module structure makes the management of module and system verification very difficult. If defects occur during the integration there can be easily situations where specifications are missing how to verify the functionality of the system.

5.2. Challenge for Designers

Dymo sets requirements for Designers and their way of working. The design procedure would be simplified from VDI-richtlinie 2221, but the design work is not eased at all. On the contrary, designers often have felt that increased constraints make theirs work more difficult when compared to “blank-paper” approach. However, this is more a matter of attitudes than a reality as discussed earlier [Juuti&Lehtonen02]. But attitudes are part of reality when facilitating change so there is challenge to change mindsets.

Some experts will need to specialise into planning the reuse of modules and their interoperability. They also provide information for decision making in roadmapping work. Other experts are needed for defining needed configurations as part of product family level planning. There are considerably many optimisation criteria to be taken into account like lifecycle of module, components etc. In some cases the designers need to use expert systems that ease the mental load and facilitate product family level planning process. Due to the impacts of the decisions in product family planning quite many people want to participate these planning sessions. Thus Decision-making takes place in groups rather than in individual minds, which is usually new situation to designers. The need for module reuse restricts using different solutions and limits the solution space. There are examples where this has an positive effect, also.

5.3. Challenge for product architecture and design management

Dymo sets crucial requirements for product architecture. Modular product structure in itself is no longer adequate prerequisite for applying Dymo. The minimum requirement is that the modules must be encapsulated with well-described interfaces. There is inherent need to make simple interfaces, with as few interactions as possible because it is inevitable that the module structure must be partially altered during the product lifetime. The more rigid and complex interactions there exist between modules, the less flexible the module structure becomes. Thus real encapsulation is needed instead of only documenting the interfaces between modules. In designing encapsulated module architecture most of the Design for Configuration (DFX) methodology rules are applicable [12].

The keeping the module structure consistent and in-line with requirements can not be done without proper management tools. So there must be module and architecture management system with a link to requirement management. These systems must be constraint-based because rule-based design support system does not posses the flexibility needed. At least during the designing phases there will be conflicting requirements and the system must be robust to enough to let the designers keep on going with the design work. Otherwise the iteration in design work and testing different solutions is not possible. Even when making deliveries out of dymo-platform, the product model might still be partially imperfect, but this should not affect working as long as delivered product does not use the modules and structures which are not ready. Since in Dymo the modules and module structures are evolving, one of the key issues is version management in multiple levels. Relating the correct versions of modules and structures with requirements is also a challenge.

We propose that Product Family Master Plan presented by DTU [13, 14] would be the most promising approach from the existing methods of documenting product families and module system. The chromosome-model presented would give guide lines in making requirement satisfaction management tools. However no readymade such tools exist at time of writing.

6. Conclusion

Platform based product development is a valid strategy for companies making variant products. In turbulent markets the platforms are problematic, because transition from an old platform to a new one is time and effort consuming process. Dynamic Modularisation solves partially the problem by making platform-based way-of-working more agile. However Dymo sets great requirements for processes, internal communication, competence of personnel and product structure and thus it is a strategy for only small portion of companies.

References

[1] Riitahuhta A., Andreasen M. M. "Configuration by Modularisation", Proceedings of NordDesign 98, KTH, Stockholm, 1998.
[2] Riitahuhta A. and Pulkkinen A. "Design for Configuration", Springer, Berlin, 2001.
[3] Pine B. J., "Mass Customization", the New Frontier in Business Competition. Harvard Business School Press. Boston, MA, USA. 1993.
[4] Tiihonen, J., Soininen, T., Männistö, T. and Sulonen, R. "Configurable products - Lessons learned from the Finnish industry.", Proceedings of 2nd International Conference on Engineering Design and Automation (ED&A '98), Integrated Technology Systems, Inc., 1998.
[5] Ulrich, K., Tung, K., "Fundamentals of product modularity", DE-Vol. 39, Issues in Design/manufacture Integration, ASME, New York, 1991.
[6] Ulrich K., Eppinger S. "Product Design and development", McGraw Hill, ISBN: 0070658110, 1995.
[7] Erixon, G., "Modular Function Deployment – A Method for Product Modularisation", Kungliga Tekniska Högskolan, Stockholm, 1998.
[8] Pulkkinen, A., Tiihonen, J., Riitahuhta, A. "Konsta Project – Design for Configuration.", Improving product development efficiency in manufacturing industries 1996-1999. Final Report, Tekes 3/2000.
[9] Hubka V. 1988., "Theory of Technical Systems: A Total Concept Theory for Engineering Design.", Edt. by W. E. Eder. Springer-Verlag. 1988.
[10]Andreasen M. M., Hansen C. T., Mortensen N. H., "On the identification of Product Structure Laws", Proceedings of 3rd WDK workshop on Product Structuring, Delft University of Technology, 1997.
[11] Juuti T., Lehtonen T. "Facilitating Change towards Platforms with Simulation Game", Proceedings of NordDesign 2002, NTNU, Trondheim, 2002.
[12] Pulkkinen, A., Lehtonen, T., Riitahuhta, A., "Design for Configuration – Methodology for Product Family development", Proceedings of ICED99, Munich, 1999.
[13] Mortensen N. H. "Desing Modelling in a Designer's Workbench – Contribution to a Design Language", Ph.D thesis, DTU, Lyngby, 1999.
[14] Pulkkinen A., Harlou U. and Mortensen N. H. "Modelling Relations in Product Families", Proceedings of NordDesign 2002, NTNU, Trondheim, 2002.

Timo Lehtonen
Tampere University of Technology
P.O. BOX 589
33101 Tampere, Finland
phone +358 3 3115 2627, fax +358 3 3115 2307, E-mail: timo.lehtonen@tut.fi