INTERNATIONAL DESIGN CONFERENCE - DESIGN 2008
Dubrovnik - Croatia, May 19 - 22, 2008.

MAINTAINING PRODUCT PLATFORMS IN INDUSTRIAL MACHINERY

A. A. Alblas and J. C. Wortmann

Keywords: product platform, engineering change management

1. Introduction

Producers of industrial machinery face nowadays a technological complexity greater than ever before. Moreover, products are continuously changing and become more and more customer driven. These phenomena have substantial effects on the way in which product development is organized; especially the way in which product platforms are designed and maintained, since platforms are a means to control variety (Erens, 1996).

The development of a platform is affected by the product complexity. Product development in idustrial machinery is characterized by an enormous quantity of engineering data, process uncertainty, frequent engineering changes and disturbances (Rouibah & Caskey, 2005). Often, no single person in the company has a detailed overview of all the systems in a product, due to the product's complexity. As a consequence, the impact of changes is difficult to predict. This holds for impact on functionality and performance and even more for the impact on costs (Eckert et al., 2004).. Assessment of possible effects of changes is mostly based on the experience of the employees (Fricke et al., 2000). However, information only coming from personal experience does not form a proper basis for decisions. Humans make errors, and the process is neither transparent and traceable, nor repeatable. Accompanying phenomena of this immature change process are uncontrolled development of variants (comparable with uncontrolled growth of variety, Erens 1996) and unstable platforms.

Most studies of platform strategies do not focus on industrial machinery, but on consumer products. In these studies, product development often aims at maximizing customer variety at minimal costs. Product platform development strategies are used to introduce this variety. Several studies in the automotive industry show that producers manage to offer high variety in a profitable way (Cusumano & Nobeoka, 1998). Comparable studies are conducted on the applicability of platforms in consumer electronics. A good example is the case of Black & Decker (Meyer & Lehnerd, 1997) and the Sony Walkman (Sanderson & Uzumeri, 1997). A common characteristic in these platform approaches is that the development of platform is a preliminary and proactive exercise focusing on the development of customer products.

The majority of platform research is focused on initial platform development, not on later life cycle phases. Obviously the development of platforms is a very important issue to focus on. However, industrial machinery, as an engineer-to-order industry, has its own characteristics and special problems in the design and the maintenance of its platforms. Successful strategies in other branches are not directly applicable in industrial machinery for two reasons. First, in industrial machinery the technological evolution of a platform is highly uncertain and difficult to define beforehand. In the production of highly innovative products, which is often the case of industrial machinery, it is difficult to define the common base and the future variants. Second, products, solutions, architectures and interfaces within the platform are continuously changing over time. Establishing a platform is not a static activity but a continuous process: the effects of changes must be continuously assessed on their impact on the platform.

The development and the maintenance of a platform is closely related to engineering change management. Accordingly, we propose to integrate engineering change management practices in the product platform lifecycle process in order to establish and maintain platforms. Consequently, this study has the following objectives. First, relevant product platform theory will be discussed with a focus on the process of setting up a platform (section 3). Second, there will be an in depth analysis of the problems of applying this platform strategy in an industrial machinery setting in (section 4). Based on this analysis we will define the requirements for product platform techniques in these type of industries (section 5). And finally conclusions will be drawn (section 6).

2. Methodology

In line with Yin (2003), a case study is used to explore the requirements for product platform processes in industrial machinery. First, the concept 'product platforms' is analyzed as well as the effects of engineering changes (EC) in platform development. Next to this, a case study is described, where data were collected by using various research instruments. We carried out a dozen interviews with open-ended questions in one case company. Interviewees consisted of personnel involved in engineering decision processes. In addition, we conducted tens of site visits over a one-year period. During these intensive site visits a large amount of minutes of meetings, procedures, design specifications, close-out reports and other miscellaneous reports where read and analyzed. Furthermore informal conversations were held and drawn up with a large variety of personnel. In the development of the requirements the case study results are compared with existing theory resulting in a new proposition for platform maintenance.

3. Product platform development processes in academic literature

In this section we will briefly discuss literature that illustrates how changes affect the difficulties in setting up and maintaining a platform strategy. Firms develop product platforms to provide sufficient market variety at minimal costs. As Simpson et al. (2004) state, "a product platform is one of the most challenging aspects of product family design". The way a platform is developed depends on the way the product development process is organized. In general, development processes are sequentially organized and can be divided into six phases, that is (1) planning, (2) concept development, (3) system level design, (4) detail design, (5) testing refinement, (6) production ramp-up (Ulrich and Eppinger, 1995). In reviewing the literature several comparable approaches can be found (e.g. Cooper & Kleinschmidt 1993, Roosenburg & Eekels 1995, Pahl and Beitz 1996). Often, a platform is developed in the start up phases of a development process. *Yet, a platform is developed for multiple products following multiple development cycles.*

Simpson et al. (2001) distinguish two common approaches in developing a platform, namely a 'top down' and a 'bottom up' approach. A top down approach is setting up a platform from scratch and a bottom up approach aims at redesigning a group of products to establish a platform. Both approaches aim at the development of a platform at some point in time. In a development process, a top down approach often starts at the beginning of the process and a bottom up approach often is executed during a product development project or when a product is redesigned. This study will cover both approaches and will look at the process flow of developing a platform (top down) and maintaining or redesigning a platform (bottom up).

In reviewing the literature, approaches can be found on both top down and bottom up development of a platform. These approaches are mainly focusing on methods to improve or to set up a platform at some point in time, without incorporating the process aspects in platform development. Very little was found on the *process* of setting up and especially maintaining a platform. The urrent scientific debate seems to focus on platform development strategies which are applicable for products that are 'frozen' after they are released for production. In industrial machinery this is often not the case. This study aims at explaining the difficulties in platform maintenance in industrial machinery production and adds to our understanding of the product platform life cycle process.

Previous research revealed a dual meaning about the term "product platform", namely an external and an internal meaning (Wortmann and Alblas, 2007). In this paper, an *external product platform* was defined as the range of offered/anticipated products in terms of functionality and performance, including choice features, customizing options, and external interfaces. The term 'external' refers to the marketing perspective on a platform. And our definition of an *internal product platform* is the variety of end products offered and the interfaces between the generic components (i.e. product architecture). With the term 'internal' we refer to the developers, producers and service perspectives. Accordingly, we define platform maintenance as *the process of performing changes to predefined platform attributes (e.g. functionality, performance, interfaces, etc.)*.

Maintaining a platform has to do with managing the process of changes and decisions on product platforms. In general, managing changes in development processes is organized in engineering change processes. The product platform life cycle process is closely related to the engineering change process. According to Jarrat et al. (2005) engineering change management can be defined as "the organization and control of the process of making alternations to products". An engineering change can be defined as a change on a product after the design is released to production. In line with this definition a platform change can be defined as a change on a platform after its release.

In academic literature several publications can be found on the management of changes in complex products (Earl et al., 2005; Eckert et al., 2004; Keller et al., 2005). These studies contributed much to our understanding of change in complex engineering domains. For example Eckert et al. (2004) describe the case of helicopter development and mention problems of change management in complex, engineering-intensive products. However, no attention is paid by the previous authors to the issue how changes in individual products should be fed back to product platforms, or how platforms should be changed. Although previous mentioned studies present valuable insights in the complexity of engineering change management, this paper intends to contribute to this literature by developing requirements of platform change processes, which pays attention to both platform development and platform maintenance after its release to production. An industrial machinery case example will analyse the problems in maintaining a platform.

4. Case analysis

The company example is a leading provider of large and complex industrial machinery. The engineering and design disciplines are highly diverse (e.g. electronics, mechanics, optics, software). The project organisation is divided into several programs aiming at several product types. The programs are divided in several projects. They are executed concurrently. Within the programs the projects, executed in parallel, are all affecting each other. Each program consists of over 200 engineers.

4.1 Uncontrolled growth of variety

In the development process customer requirements are changing, not only during development phase, but even during the production and maintenance phases. This leads to a continuous flow of technological change during the whole product life cycle. In addition the engineering work has often to do with reuse of existing artifacts from previous platforms. Changes of, and additions to, the existing platform form the basis for the next generation products. As a consequence, the products are never developed from blank sheet level: there is always a design history. Surprisingly, no method was available to extract a product from a product platform. Interviews reveal that the platform design is based on the carrier product: the first product developed on in a new platform range. After the release of this platform the subsequent products are developed based on this design. Principles found in the new product are not fed back to the platform design. So formally there is no platform baseline.

Engineering decision making is highly complex in the company example. A major source of complexity is the large amount of people involved in the engineering organisations which makes project management complex. Interviews show that current platforms are difficult to stabilize and maintain. At the starting point of a new development the engineering departments tries to set up a generic design of a series of machines with several variants. An interesting finding was that no common methods were available to maintain the stability and commonality of the platform. Designers are changing their designs without having insight into the platform impact. As a result, many unplanned variants grow and each product becomes one-of-a-kind. In the company the product architecture is complex and continually changing. Effects of these changes are hard to predict. Therefore, establishing a product platform is a complex process.

4.2 Platform change vs product change

In our studies we followed several initiated changes. Figure 1 illustrates a simplified example with two engineering changes. In this situation the effect is that the platform design slowly differentiates from the product design resulting in a vague parent-child relation.

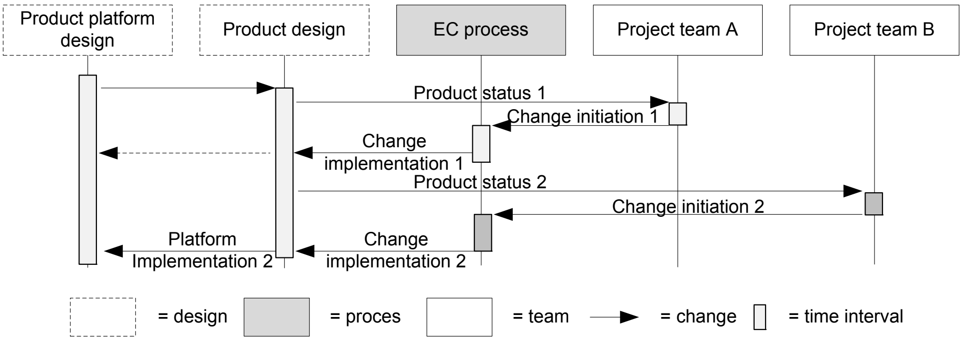

Figure 1. Simple change implementation sequence

The initiated change is based on a platform which is in the case example the same as the carrier product. To assess the technological impact of a change the status of a design is extracted by the project team. The team addresses a change to a change control board positioned in the EC assessment phase of the EC process. After approval the change is implemented in the product. In the case study not all the changes are implemented in the platform design. The dotted arrow illustrates that the change initiated by project team A is not implemented in a platform design. However, the change of project team B is not only implemented in the product design, but also in the platform design. Interviews show that these decisions whether to implement a change in a platform design are not made rationally. This results in further separation of the two designs.

Note the change arrows sketched in the figure are just example changes and do not represent the number of engineering changes. In this company example over thousands of engineering changes are implemented during the last years. Even during one development hundreds of engineering changes can occur. Thereby, innovations and solutions to problems found in one product development must often be implemented in other product developments.

4.3 Parallelism of changes

In a simplified version of the engineering change process of the case example a change has the following lifecycle: initiation, approval and implementation. Changes can also occur in parallel. Our study reveals that the variance of throughput time of a change is high and changes can be initiated in parallel. As a consequence, changes can be based on different product states and the implementations can be unreliable because of changing circumstances. The next figure shows a simplified example of two parallel changes.

Project team A wants to change an certain part of the product design. For this reason information is needed about the status of the product design. Based on this a change proposal is developed and presented to the change control board. After acceptance the change is implemented in the product design. This results in a new product status or even a new product version. In parallel project team B follows the same procedure in order realise another improvement on the same product. This change is implemented in a product with the same status as product status 1 without taking into account the influence of change 1 on the functionality of change 2. In this situation change 1 is passed by change 2 causing unpredictable changes. So even in this simplified situation it is possible that the relationship between effect of change 1 and the effect of change 2 is uncertain and not taken into account. In reality there is an ongoing process of changes where the mutual interaction is very unpredictable.

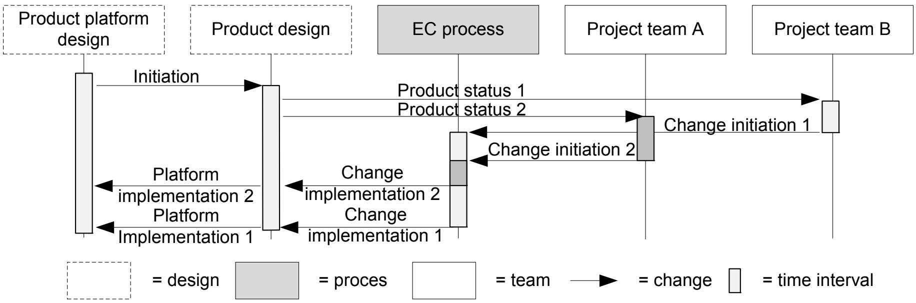

Figure 2. Parallel change implementation sequence

To prevent uncontrolled change implementations, coordination is needed between the different innovations. Therefore, a clear picture of the state of a product during the initiation and the implementation of a change is important. A similar phenomenon can be observed in health care: also there it is rather important to overview the whole set of therapies in steed of looking to a single medicine. As a consequence, the throughput time and implementation moment of a change are important elements to take into account in managing a product platform. Especially when a product is continuously changing over time.

This situation becomes even more complex when several product developments are executed in parallel. Platform change decisions do not only affect one innovation but multiple innovation cycles. At the company example multiple designs are executed in parallel, so impact assessment must assess the impact of a change on multiple design cycles. Interviews with members of the change control board reveal that is quite often difficult to define the range of impact a change. It is difficult to assess whether a change is applicable to product A, product A and B or the whole platform. Next to it, problems arise in defining whether a change must be compatible with previous or future variants.

4.4 Difficulties in platform development in embedded architectures

In the case study, it is clear that the continuous stream of changes on products makes maintaining a product platform complex and uncertain. Moreover, these changes are in conflict with the attempt of firms to control their decision processes in order to attain predictable and stable platforms. To realise predictability and stability Suh (1990) propagates the importance of decomposing problems into a problem hierarchy. Accordingly, Erens (1996) defines the decomposition step in dividing the overall problem in sub-problems. In the development of products with an integral architecture this decomposition process is difficult. A high level of function sharing and geometric nesting causes difficulties in realizing a modular structure (Ulrich, 1995). As a consequence, interactions between project decisions and platform decisions (e.g. on reusable modules) are vague and difficult to manage. This interferes with the issue of transforming functional problems into technological solutions. As a consequence the effects of platform change decisions are difficult to determine. If we compare automotive industries with industrial machinery, the platform in automotive industry does not change fundamentally after its release to production, whereas in industrial machinery, the platform lacks stability. As stated before, in industrial machinery the end-products continually change over time. Because of the architectural complexity, the impact of change is unpredictable, which causes change propagation effects. Hence, change propagation affects the stability of the platform and the predictability of platform decisions. Continually changing requirements and solutions makes the environment of the decision problem uncertain. For example when a platform is developed it is difficult to predict which components will be changed over time.

In our case example, decisions are difficult to assign to a platform, i.e. a set of products or individual configurations. There is no structured method to allocate design problems to a series of products or even generations of products (product platform): in the company example the engineering change management process affects the development of a platform. In general, therefore, the following problems in product platform development can be identified, namely:

- decomposing the problem into sub-problems is complex, which makes establishing a stable platform architecture difficult;
- a product environment where products are intensively and continuously changing, even in their manufacturing and service phase, makes maintaining a product platform complicated;
- difficult to feed back changes to the platform without a platform base line;
- many parallel changes with different implementation times makes implementation of changes complex;
- The implementation range of an EC is often difficult to define.

Based on the problems mentioned above requirements will be developed for product platform maintenance in industrial machinery in the next section.

5. Requirements for product platform maintenance

In the previous sections we have analysed literature and a case example. The presented examples show that in companies like the case example engineering change management is closely related with platform development. Questions like ‘Is the platform affected by this change?’ and ‘Is this change the next standard?’ are crucial. Accordingly, not only platform planning and development, but also platform maintenance is essential. As stated before, the engineering change process is at the heart of the question whether or not to change a platform. So, for platform maintenance change criteria are needed. The first step in structuring this decision process is categorizing the engineering changes based on their impact on the platform. This section will present some first steps in making this categorisation. Next to this, at the end of this section we will present three phases in the platform development process.

5.1 Change categorisation based on change impact

This study has found that the products are continually changing and that there are many engineering changes during multiple development life cycles. This continuous flow of implementation of ECs on different products makes stabilizing a product platform extremely difficult. Sometimes changes have to be implemented in a range of products and sometimes they are product specific. Accordingly, we need to distinguish between changes based using their impact range. As a result, the following distinction between the effects on individual products, multiple products or a product platform must be made. Table 1 below summarizes this distinction based on the internal en external platform definition given in the introduction resulting in an internal and external question.

Table 1. Impact of a change

Individual product life cycle affected	Multiple product life cycles affected	Platform life cycle is affected
▪ Internal question: Does this change affect only one development? ▪ External question: Is the change a customer specific solution?	▪ Internal question: Does this change affect multiple development cycles? ▪ External question: Is the change a solution for multiple customers?	▪ Internal question: Is this change the standard for current platform? ▪ External question: Is the change a solution for all the customer solutions within the product range?

5.2 Timeliness of a change as discriminating factor

An additional important characteristic that needs to be made is the distinction between upward and downward effects and the distinction between current and next life cycle. Crucial in assessing changes is its compatibility with previous or future variants of a product or a product platform. The technological impact assessment of this change is often based on the impact on the change carrier, i.e. the product where the problem is found. Quite often these changes are implemented in multiple products or even in multiple platforms. So, the assessment of the effects of these changes can differ based on their upward or downward compatibility with previous or future variants of a product or a product platform. The next figure illustrates this distinction.

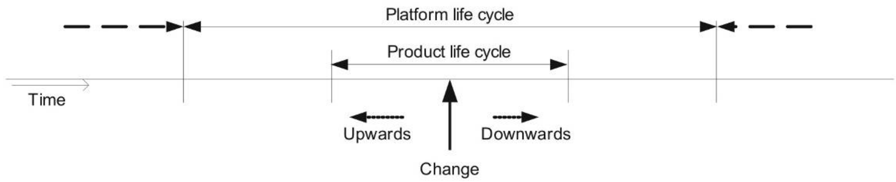

Figure 3. Timeliness of a change

Also the implementation of a change does not always have to be on current product or product platform. In several cases the change is feasible, but not on current platform. Reasons can be that current platforms are frozen and the costs of this change are too high to implement on this platform. Also, the phase of a platform in its life cycle is essential. According to the 'Rule of Ten' the costs of a implementing a change become ten times higher with each subsequent phase (e.g. Clark and Fujimoto, 1991). In this situation engineers can decide to plan the change for next platform. Therefore, in a change assessment the development phase of the artifact is crucial, especially in the development of a platform.

An important consideration concerning timeliness of changes is the availability of resources in product development. In many cases where a change would preferably be implemented in a product and in the platform with upward and backward compatibility, shortage of resources enforces another decision. This phenomenon was also observed in the case study.

5.3 Description of platform change attributes

In section 3 we defined platform maintenance as the process of performing changes to platform attributes (e.g. functionality, performance, interfaces, etc.). The case study revealed that because of the difficulty to decompose the a problem into several sub-problems in industrial machinery, maintaining a product platform is complex. It is highly likely that a product platform will change over time. Therefore, the allowed change attributes to a platform must be specified in the platform definition phase. Change management of a platform must be based on these attributes. Based on the distinction between the internal and the external platform definition made in section 3 we can define change attributes that may be affected when changing a platform. External platform attributes could be functionality and performance. Internal platform attributes could be interfaces, modules and the architecture. The effects of a change on these attributes determines the impact of a change on the platform.

5.4 Platform change elements and platform maintenance management

Table 2 summarizes the most important elements in platform maintenance (i.e. platform change management).

Table 2. Elements in platform maintenance

Impact range of a change	Timeliness of a change	Affected platform attributes
▪ Individual product life cycle affected ▪ Multiple products life cycles affected ▪ Platform life cycle is affected.	▪ Downwards/ upwards compatible ▪ Current/ next platform ▪ Phase of a platform in its life cycle	▪ Functionality ▪ Performance ▪ Interfaces ▪ Modules ▪ Architecture

In previous sections we identified the importance of platform attributes and platform maintenance management. Essential is the difference between *defining* a platform, *planning* a platform and platform *maintenance*. Based on our study we can make this distinction. Accordingly, we propose the distinction of the following phases in the platform development process:

- Phase 1: Platform definition phase
  In this phase the attributes of the platform need to be defined in terms of functionality, performance, interfaces and modules. Next to this, the platform architecture must be defined. Moreover, it has to be specified, which attributes are considered subject to change in the context of this platform
- Phase 2: Platform planning
  In this phase the lifecycle of the platform, including the milestones, needs to be defined. Also a policy on the attributes in relation with these milestones needs to be developed.
- Phase 3: Platform maintenance
  In this phase the attributes are under change control and all the changes that affect the attributes of the platform need to be assessed on their platform impact.

Consequently, we can define four questions that are essential in the platform maintenance phase, namely:

- Are the platform attributes affected?
- Will the change be implemented in current or next platform?
- Must the change be downwards or upwards compatible?
- What is the phase of the platform in its life cycle?

As a result, implementation of a platform maintenance method will stimulate rational decision making on platforms and can be considered as a novel piece of the puzzle in platform development.

6. Conclusions and further work

This paper has argued that platform maintenance is essential for dealing with the complexity of industrial machinery. Setting up a platform without incorporating the process aspects of platform development is not enough: platform development is a continuous process. In industrial machinery the technological evolution of a platform is highly uncertain and difficult to define beforehand. Maintaining a platform in an environment where products are intensively and continuously changing is extremely difficult. Based on our study we identified several important elements and defined some possible platform attributes. Further research could focus on defining the attributes of a platform more precisely and on the development of a change or freeze policy on these attributes. Also, we gave an overview of elements that seem important in categorization of changes and we have distinguished three phases; platform definition, platform planning, and platform maintenance. Essential in these phases is the definition, the planning and the maintenance of the platform attributes. The findings of this study add substantially to our understanding of platform development in industrial machinery. However, further work needs to be done on filling in the details of a platform maintenance method and its relation to engineering change management processes. Next to it, for external validity, work needs to be done on the comparison of the case study results with other companies.

References

Clark, K.B., Fujimoto, T., "Product development performance", Harvard Business School Press, 1991.

Cooper, R. & Kleinschmidt, E., "Stage Gate Systems for New Product Success", Marketing Management Vol.1, No.4., 1993, pp 20-9.

Cusumano, M. & Nobeoka, K., "Thinking Beyond Lean: How Multi-project Management is Transforming Product Development at Toyota and Other Companies", Free Press, 1998.

Earl, C.; Eckert, C. & Clarkson, J., "Design change and complexity", 2nd Workshop on complexity in design and engineering, 2005.

Eckert, C.; Clarkson, P. & Zanker, W., "Change and customisation in complex engineering domains", Research in Engineering Design Vol.15, No.1., 2004, pp 1-21.

Erens, F., "The Synthesis of Variety, Dissertation", Eindhoven University of Technology, 1996.

Fricke, E.; Gebhard, B.; Negele, H. & Igenbergs, E., "Coping with changes: Causes, findings, and strategies", Systems Engineering Vol. 3, No.4., 2000, pp 169-179.

Jarratt, T.; Keller, R.; Nair, S.; Eckert, C. & Clarkson, P., "Visualization Techniques for Product Change and Product Modelling in Complex Design", Diagrammatic Representation and Inference: Third Intational Conference, Diagrams 2004, Cambridge, UK, 2004.
Keller, R.; Eckert, C. & Clarkson, P., "Multiple Views to Support Engineering Change Management for Complex Products", Third International Conference on Coordinated and Multiple Views in Exploratory Visualization, 2005, pp. 33-41.
Meyer, M. & Lehnerd, A., "The Power of Product Platforms: Building Value and Cost Leadership", Free Press. 1997.
Pahl, G. & Beitz, W., "Engineering Design: A Systematic Approach", Springer, 1996.
Roozenburg, N. & Eekels, J., "Product Design: Fundamentals and Methods", Wiley, 1995.
Rouibah, K. & Caskey, K., "Change management in concurrent engineering from a parameter perspective", Computers in Industry Vol.50, No.1., 2003, pp 15-34.
Sanderson, S. & Uzumeri, M., "Managing product families", Irwin, 1997.
Simpson, T.; Maier, J. & Mistree, F., "Product platform design: method and application", Research in Engineering Design Vol.13, No.1., 2001, pp 2-22.
Suh, N., "The Principles of design", Oxford University Press, 1990.
Ulrich, K. & Eppinger, S., "Product design and development", McGraw-Hill New York, 1995.
Ulrich, K., "The role of product architecture in the manufacturing firm", Research Policy Vol.24, No.3., 1995, pp 419-440.
Wortmann, J., Alblas, A., "Product platform life cycles: a multiple case study" International Journal of Technology Management, 2008, in press.
Yin, R., "Case Study Research: Design and Methods", Sage Publications Inc, 2003.

A.A. Alblas
PhD Student
University of Groningen, Department Business & ICT
Landleven 5, Groningen, PO Box 800, 9700 AV, The Netherlands
Tel.: +31 50 363 3864
Fax.: +31 50 363 2275
Email: a.a.alblas@rug.nl