Analyzing NPD projects from an operational control perspective

Citation for published version (APA):
Oorschot, van, K. E. (2001). *Analyzing NPD projects from an operational control perspective*. [Phd Thesis 1 (Research TU/e / Graduation TU/e), Industrial Engineering and Innovation Sciences]. Technische Universiteit Eindhoven. https://doi.org/10.6100/IR549472

DOI:
10.6100/IR549472

Document status and date:
Published: 01/01/2001

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 16. Nov. 2023

Analyzing NPD Projects from an Operational Control Perspective

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Oorschot, Kim E. van

Analyzing NPD projects from an operational control perspective / door
Kim E. van Oorschot. – Eindhoven : Technische Universiteit Eindhoven, 2001.
– Proefschrift. -
ISBN 90-386-0983-3
NUGI 684
Keywords: product development / NPD / radical (new) product development / project management / process control / uncertainty / decision processes

Druk: Universiteitsdrukkerij, Technische Universiteit Eindhoven

Analyzing NPD Projects from an Operational Control Perspective

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de Rector Magnificus, prof.dr. R.A. van Santen, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op dinsdag 9 oktober 2001 om 16.00 uur

door

Kim E. van Oorschot

geboren te Rotterdam

Dit proefschrift is goedgekeurd
door de promotoren:

prof.dr.ir. J.W.M. Bertrand
en
prof.dr. C.G. Rutte

"Henk
Wie mag dat wezen?"

(Acda & de Munnik, *Henk*, 1997)

Preface

The work that started a little over four years ago is finished! Only a couple of months before I received my Master's Degree, I realized that perhaps there was a researcher in me. Corné Dirne talked me into it. He tried to persuade me for a research project by saying that as a researcher you have to like three things: reading, thinking and writing. He did not need to say anymore. I was convinced. Together with Ton van de Wakker from ASML, Corné developed a research proposal that I found interesting and that Will Bertrand found promising. My project could begin.

In addition to doing the things I liked, I had a great time during the research project thanks to the colleagues at the LBS department. My roommates Joris Keizers and Inge Satter always made me laugh, and sometimes made me coffee. For me these are necessary ingredients for doing research. Furthermore, Willem de Paepe, Ebbe Negenman, Nico Dellaert and the 'old gentlemen' of the KRR made my stay at the university very nice, particularly by *not* talking about research.

Besides the colleagues at the university, I would like to thank the colleagues of the PPC department at ASML, especially Ton van de Wakker, who where always willing to cooperate.

And of course a special thanks goes out to my supervisors Will Bertrand and Christel Rutte who both motivated and inspired me throughout these four years.

Finally, I would like to thank my parents and my friends, in particular Sierd Hoekstra and Ron Bisschops, and of course Lysbeth Slotegraaf and Jolanda van Bergeijk, who provided me with the motto for my thesis.

Kim van Oorschot,
May 2001

Contents

1 Introduction and problem statement

2 Characteristics of radical NPD projects

3 Planning and control of radical NPD projects

4 ASML, case description

5 Three small-scale empirical studies

6 Empirical study of radical NPD projects

1 Introduction and problem statement

1.1 Introduction

In developing new products, time-to-market (i.e. the time in which the new product is developed, manufactured and ready for sale) is an important measure. Time-to-market used to be determined by the specifications of the product, the (financial and human) resources available for developing the product, and the functional organization of the product development process. In the nineties, time-to-market became more and more a performance parameter of the new product. A parameter for a customer to select one product instead of another competing product and thus a parameter for product development firms to create competitive advantage. Nowadays, a short time-to-market in some businesses has become so important that it is almost more appropriate to call it *speed*-to-market. This is especially true for products in the high-technology sector or in the e-business, where the deadline or due date at which a new product must be ready is set even before the specifications of the product are defined completely. In this situation new product development (NPD) is time-constrained and the objectives of new product development processes have changed. Traditionally, product development objectives were defined in terms of realizing a given product quality or given product specifications, while minimizing development costs and/or minimizing development time. However, in time-constrained NPD projects, objectives are defined in terms of maximizing product quality or maximizing the extent to which product specifications are realized, within a given time constraint. This change in objectives might also change the way in which development processes are planned, executed, and controlled. The planning, execution and control of these processes is the topic of this thesis. To explain the focus of this thesis, it is important to distinguish two types of time-constrained NPD projects: incremental and radical NPD projects (McDermott, 1999). In incremental NPD projects the development processes pertain to small, simple changes in existing products. Here, the product quality or specifications that can be realized within a given time constraint can be planned accurately in advance, because the technological uncertainty is low. Consequently, it is plausible that techniques that are used to plan and control 'traditional' NPD projects (in which product quality was a given and time-to-market was a derivative) are also suitable to plan and control incremental NPD projects. In radical NPD projects completely new products or platforms are developed. Here, technological uncertainty is high and it is very difficult to predict and plan the product quality or specifications that can be realized within the given due date. Because of the high technological uncertainty, radical NPD projects seem to differ more from the 'traditional' projects. It is expected that traditional planning and control techniques are not suitable for time-constrained radical NPD projects. Therefore in this thesis we focus on the planning and control of time-constrained radical NPD projects.

Little is known about the characteristics of the technological uncertainty and the way it influences the execution of time-constrained radical NPD projects. Previous operational research is directed on the one hand to 'the fuzzy front end' of projects, in which project objectives are set and project teams are formed, instead of the execution phase of projects (Tatikonda and Rosenthal, 2000). On the other hand, previous operational research addresses planning and scheduling algorithms, taking into account precedence and/or resource constraints and an objective function in terms of time or costs (see for example Davis and Patterson, 1975; Kurtulus and Davis, 1982; Özdamar and Ulusoy, 1995). These algorithms are based on a number of assumptions regarding the project to be planned and controlled and it can be questioned whether these assumptions hold for time-constrained radical NPD projects. Because of this, we will examine the characteristics of the execution phase of time-constrained radical NPD projects. When these characteristics are known, we will evaluate whether the assumptions of the existing planning and scheduling algorithms hold and we will suggest requirements for the planning and control of time-constrained radical NPD projects.

The outline of the remainder of this chapter is as follows. In Section 1.2 the motivation of this research is described. In Section 1.3 the problem statement and research questions are formulated, followed in Section 1.4 by the research methodology that we adopted. Finally, in Section 1.5 the outline of this thesis is given.

1.2 Motivation of the research

To explain the motivation of the research, two different viewpoints are distinguished: motivation from a scientific point of view and motivation from a managerial point of view. These will be described below.

Motivation from a scientific point of view
Planning and control of time-constrained radical NPD projects seems to be influenced strongly by the technological uncertainty of these projects. Because of this uncertainty, it seems not possible to predict accurately the functional product specifications that can be realized at the due date of the project. Nevertheless, as was discussed in the previous section, product development literature is focused more on the fuzzy front end, where specifications are defined instead of actually executing the project (Bhattacharya *et al.*, 1998; Tatikonda and Rosenthal, 2000; Repenning, 2000). Furthermore project planning literature is focused more on developing planning and scheduling algorithms that minimize development time or cost instead of maximize product quality (Boctor, 1990; Kolisch *et al.*, 1995; Icmeli Tukel and Rom, 1998). Therefore, from a scientific point of view we are interested in the execution phase of time-constrained radical NPD projects and the way the development processes in this phase can be planned and controlled.

Additionally, existing planning and scheduling algorithms seem to ignore that development processes are planned, performed, and controlled by humans: engineers and project leaders. Therefore, besides examining the characteristics of development processes, it seems also important to examine the behavior of the people involved in these processes. Because engineers and/or project leaders make the project plans, research of their ability to make estimates of processing times of work packages is useful here. In addition, work packages in radical NPD projects involve problem solving skills of engineers. Therefore research of problem solving processes can be suitable here as well. Furthermore, in time-constrained radical NPD projects, the due date of the project is of the utmost importance. The nearer the due date, the higher the time pressure. Thus, research of how people behave under pressure or how their behavior changes as pressure increases is important as well. These three topics described here, have been studied and described already in psychological literature. The scientific motivation emerges, because we intend to use psychological knowledge in our analysis of time-constrained radical NPD projects. These projects will be analyzed from a perspective in which psychology and operations management research are combined. The combination of these two knowledge areas is not common and therefore interesting from a scientific point of view.

Motivation from a managerial point of view

Besides the scientific interest, examining the planning and control of time-constrained radical NPD projects is interesting also from a managerial point of view. For many firms the execution phase of NPD projects is a struggle (Repenning, 2000; Wheelwright and Clark, 1995). Realizing the project due date is extremely difficult in many projects. When the due date approaches, either long delays or an increase of allocated resources are reported. Buehler *et al.* (1994), for example, described construction projects that had delays of 10 (the Sydney Opera House) or even 13 years (Olympic Stadium of Montreal). Icmeli Tukel and Rom (1998) performed a survey among project managers in various industries. 52% of the 90 respondents reported that projects were often delayed and 41% reported that projects were sometimes delayed. These project managers also reported that they used project planning techniques, but this did not prevent the projects from running late. Besides these results found in literature, it is not hard to find examples of projects running late in real life. Research projects in the academic world also have similarities with radical NPD projects. And also in this environment we see projects exceeding their due date, or researchers having to work overtime to realize due dates. Performing radical NPD projects in order to maximize product quality within the time given seems to be very difficult considering all examples that can be found in real life. Therefore examining this problem and contributing to improve the planning and control of these projects is interesting from a managerial point of view.

1.3 Problem statement and research questions

The research presented in this thesis is initiated by the following problem statement:

> *What are the characteristics of time-constrained radical NPD processes and how should these processes be planned, performed and controlled in order to maximize product quality within a given lead time and with a given (limited) available capacity?*

In trying to solve this problem, it is broken down in three research questions that will be answered in this thesis. These research questions are:

1 What are the operational characteristics of time-constrained radical NPD processes?
2 To what extent can existing planning and control techniques be used for time-constrained radical NPD processes?
3 What are the additional requirements for the planning and control of time-constrained radical NPD processes?

The methodology that is used for answering these research questions is described in the next section.

1.4 Research methodology

In this research radical NPD processes are analyzed from an operational control perspective. In this section we will first explain what we mean by an operational control perspective. Second, we will describe the steps taken to perform this research.

Operational control perspective

The first use of the name Operational Research (OR) occurred in 1937, but the OR activity did not really develop until the Second World War when effective operations were of great significance. According to Beer (1959) Operational Research is 'the attack of modern science on problems of likelihood (accepting mischance) which arise in the management and control of men and machines, materials and money in their natural environment; its special technique is to invent a strategy of control by measuring, comparing and predicting probable behavior through a scientific model of a situation' (pp. 16-17). A few terms in this definition need special attention: OR solves problems in *management and control* of men and machines, materials and money in *their natural environment*. So, OR serves operations management by providing scientific solutions. Before the scientific knowledge is applied by operations managers, they must understand and accept it. Therefore, the scientific knowledge must relate to the perception of the problem situation in *its natural environment* held by these managers. Keys (1991) supports this view by writing that model-building activities in OR consists largely of the construction of a formal model of a real-world situation. In other words, when solving an OR problem two situations must be mapped, a managerial and a scientific situation, perceived by management and scientist respectively. This is why a large part of this thesis is devoted to describing an in-depth empirical study, in which time-constrained radical NPD processes are examined in their natural environment. Furthermore, for understanding this thesis it is important to realize that we analyze NPD projects from an operational control perspective. This means that we are interested in how these projects are or might be controlled and we study only those factors that can influence or are influenced by the way projects are controlled. For example, the actual process of product development (that is converting functional product specifications into technical product specifications) is not the object of study. We will neither study the mechanical calculations that are used, nor do we examine the design of the product architecture. What an engineer actually does when he develops something is considered as a black box. We are only interested in information about the progress made in this black box that can be used for controlling the development process, (for example the remaining workload or the progress of an engineer estimated in man-hours) and in controlling information for this black box (for example the remaining time until the due date).

Methodology
The methodology describes the steps that are taken to perform this research. In general, three steps can be distinguished here.

1 In the first step previous research of new product development and of human behavior was analyzed to identify characteristics of time-constrained radical NPD projects. Furthermore, traditional project planning and control techniques were examined to find the assumptions of real life projects on which these techniques are based.
2 In the second step we studied our research problem, described in the problem statement, in a natural environment. Namely we studied real life time-constrained radical NPD projects in a product development firm.
3 In the third step characteristics of time-constrained radical NPD projects resulting from the literature and the empirical study were used to derive requirements for the planning and control of these projects. Based on these requirements a radical NPD project planning and control structure was proposed.

In the next section we describe in which chapter of this thesis the results of these three steps can be found.

1.5 Outline of the thesis

The remainder of this thesis is organized as follows. In Chapter 2 operational characteristics of time-constrained radical NPD projects are identified based on a literature study. Furthermore, characteristics of traditional project planning and control techniques are discussed. Chapter 3 describes first how radical NPD projects are planned when a traditional planning technique is used. Second some characteristics of human behavior are discussed that have an important influence on the way radical NPD projects are performed and controlled. In Chapter 4 the product development firm in which the empirical studies are performed is introduced. In Chapter 5 the results of the three small-scale empirical studies are discussed, followed by the results of the in-depth empirical study in Chapter 6. In Chapter 7 the requirements for controlling time-constrained radical NPD projects are derived, and a control structure is proposed. Finally, in Chapter 8, conclusions are drawn and suggestions for further research are given.

2 Characteristics of radical NPD projects

2.1 Introduction

In this chapter results are described of a literature study of the characteristics of radical new product development processes and of the planning and control of these processes. In Section 2.2 product development processes are defined and a typology of product development processes is given. This research is focused on radical New Product Development (NPD) processes as opposed to incremental NPD processes. Uncertainty that characterizes radical NPD processes is discussed in Section 2.3. In Section 2.4 implications of uncertainty for controlling radical NPD processes are discussed.

2.2 Product development processes

Ulrich and Eppinger (1995) define the *new product development process* as: 'the sequence of steps or activities that an enterprise employs to conceive, design, and commercialize a product' (pp. 14). Numerous researchers adopt a project perspective when examining product development processes, because managing the development of an individual product can be viewed as the organizational process of managing a project (see for example Brown and Eisenhardt, 1995; Clark and Fujimoto, 1991; Taylor and Moore, 1980; Tatikonda and Rosenthal, 2000). A project is usually defined as a one-time activity with a well-defined set of desired end results or goals (see for example Meredith and Mantel, 1989; and Shtub *et al.*, 1994). A project can be divided into subtasks that must be accomplished in order to achieve the project goals. A project is sufficiently complex in order to require careful planning and control of subtasks in terms of timing, precedence, cost, and performance. The project itself must often be coordinated with other projects carried out by the same parent organization (Meredith and Mantel, 1989). A product development project is a combined effort of all organizational functions: marketing, development, purchasing, manufacturing, sales, and finance. On a high aggregate level, five generic phases are distinguished for the development function (Ulrich and Eppinger, 1995):

- concept development: establish target product specifications, investigate feasibility of concepts;
- system-level design: generate alternative product architectures, define major subsystems and interfaces;
- detail design: define part geometry, assign tolerances, and choose materials;
- testing and refinement: perform reliability -, life -, and performance tests, implement design changes;
- production, ramp-up, and launch: evaluate early production output.

On an even higher aggregate level some authors divide the product development project in two major phases: the 'fuzzy front end' or the project planning phase on the one hand and project execution on the other hand (Tatikonda and Rosenthal, 2000; Smith and Reinertsen, 1995; Ulrich and Eppinger, 1995; Wheelwright and Clark, 1995). The project planning phase includes identifying customers and competitive products, generating ideas, choosing the project to work on, setting product and project targets, and detailed network scheduling. The project execution phase involves actually carrying the product development project through completion. Tatikonda and Rosenthal (2000) note that substantial operations management literature exists on the topic of project planning. But the project execution phase of product development projects has received little attention in this literature, which has instead focused primarily on detailed network scheduling approaches used in the project planning phase. The current research is focused on the execution phase of product development projects. In this research a product development *project* refers to the execution phase and a product development *process* refers to the development processes that are performed within the project in this execution phase.

When product development projects are examined on a more detailed level, different types of projects can be distinguished. See for example Andreasen and Hein (1987), Clark and Wheelwright (1993), Turner and Cochrane (1993), and Reiss (1996). Three typologies are discussed here:

- Engineer-to-order versus customer-order *in*dependent development. In engineer-to-order product development, the development process is initiated by a customer order, in which the customer specifies what has to be developed. When product development is independent of a customer-order, the development process is initiated by the development organization (based on market research). The specifications of the new product are then defined by this development organization.
- Incremental versus radical product development. In incremental product development only small changes of existing products are permitted. In radical product development completely new products or platforms are developed.
- Compression versus experiential product development. Compression product development is considered as a certain (incremental) product development process with known development goals, and of which the development time can be shortened by compressing development tasks. Experiential product development is surrounded by uncertainty; at the start of the development project it is not known which or the extent to which functional product specifications will be realized at the end of the project.

Engineer-to-order development was examined by Muntslag (1993) and Hanssen (2000). The development process in engineer-to-order production starts with the acceptance of the customer order and the assignment of price and due date. To fulfil the agreements made with the customer and the goals of the production organization, prices and due dates must be based on realistic estimates of cost price and lead time. Realistic estimates can be made only if uncertainties of both customer order (the objective of development processes) and development processes are low. Uncertainty in engineer-to-order environments can be expressed in terms of what part of the product is pre-defined (or standardized). Muntslag (1993) distinguishes five Order-independent Specification Levels (OSL). The higher the pre-defined specification level, the less uncertainty is present in the customer order or in the objective of the development process. When a customer order is accepted with a low OSL there will be a high risk of not realizing the project within the specified due date and budget, because of the high uncertainty. Therefore engineer-to-order projects are suitable for incremental product development projects, with low uncertainty. This is consistent with the research of Hanssen (2000) who found that only minor innovations are applied in engineer-to-order development projects with low uncertainty. In incremental product development projects only slight changes of the product or processes are applied. Current product offerings or existing processes are extended. Radical innovations involve the development or application of significantly new technologies or ideas into markets that are often non-existent (Ettlie *et al.*, 1984; McDermott, 1999). Radical New Product Development (NPD) goes far beyond the OSL's of Muntslag. In radical NPD the objectives of the development process are uncertain, and it is even uncertain which development processes have to be performed to realize these objectives. Radical product development is surrounded by much more uncertainty than incremental product development and therefore it is extremely difficult to estimate accurately the time and money required to fulfil initial functional product specifications. It is even not certain whether initial functional specifications *can* be fulfilled at all. When product development projects can be performed independent of any customer order they are more suitable for radical NPD projects, because then initial functional specifications can be changed (higher/lower, more/less, etc.) during development, since no hard agreement is made with a customer. Therefore, at the start of the project more development risk can be taken, and more new technologies can be explored.

In this respect Eisenhardt and Tabrizi (1995) distinguish two models of product development. The first model is equivalent to incremental product development and leads to a product development project consisting of certain, well-defined and stable processes or tasks. Once all design tasks have been performed, the product can be made exactly according to its original planned specifications. The amount of time and resources required for each task is not uncertain. A number of tasks can be performed simultaneously, thereby 'compressing' the project duration. Eisenhardt and Tabrizi describe this type of project as *the compression model of product development*. The second model of product development, which is related to radical product development, leads to a product development project consisting of uncertain, ill-defined, and unstable processes or tasks. They call this *the experiential model of product development*. At the start of the project it is uncertain *which* tasks are really necessary for realizing the functional specifications and it is even uncertain *which* (or the extent to which) functional specifications can be realized within a certain time period. In these kinds of development projects the tasks are often reorganized during execution and the functional specifications of the product are reconsidered.

The distinction between incremental versus radical, compression versus experiential, and customer-order dependent versus customer-order independent product development is strongly related to the amount of uncertainty. Therefore, in the next section we examine uncertainty in product development projects in more detail.

2.3 Uncertainty in product development projects

Galbraith (1973) defines uncertainty as 'the difference between the amount of information required to perform a task and the amount of information already possessed by the organization' (pp. 5). If a task is not completely understood before it is started, then during the execution of this task more knowledge is acquired which leads to changes in resource allocations, schedules, and priorities. According to Galbraith these changes require information processing during task performance. The higher the uncertainty, the more information needs to be processed during task performance. Kidd (1991) writes that each task in a project carries three forms of uncertainty: uncertainty in time, cost and performance. Shtub *et al.* (1994) describe the same triple. Uncertainty in time or scheduling can be the result of changes in the environment that are impossible to forecast accurately at the outset of a project and that are likely to have a critical impact on the length of certain tasks. Uncertainty in cost can be the result of limited information on the duration of tasks that makes it difficult to predict the amount of resources needed to complete the tasks on schedule. Furthermore, the hourly rate of resources and the cost of materials used to carry out project tasks may also be variable. Uncertainty in performance or technology can be the result of developing or employing new technologies, methods, equipment, and systems. Sander and Brombacher (2000) introduce a fourth form of uncertainty: uncertainty in information. Uncertainty in information arises when innovation speed is high, time-to-market is short, and engineers are developing a new generation of products without knowing whether the quality of the current generation is satisfying. According to the authors this form of uncertainty is strongly related to the measure in which information from the production and sales department (and the customer) is communicated to the development department. Since we focus our research on the execution phase of radical NPD projects (thus excluding production and sales), uncertainty in information is beyond the scope of our research.

Uncertainty in technology may affect the schedule, the cost, and the ultimate success of the project, and thus uncertainty in technology may also increase the uncertainty in scheduling and cost. Especially in radical NPD projects high uncertainty in technology exists. However, it is surprising that much existing research on product development focuses on the upfront identification of customer needs or concept development phase and on the downstream task of realizing a product given its specifications (Bhattacharya *et al.*, 1998). In other words, it is assumed that no or little uncertainty in technology exists and that initial product specifications can be realized at the end of the project. Bhattacharya *et al.* explain why early product definition, recommended as a good practice in the literature, is not always a desirable approach. In such an approach one ignores: the uncertainty surrounding product specifications, the firm's risk aversion, and the ability to benefit from customer input during the definition process. Recent literature shows, however, that more and more researchers are examining the variability of product specifications. Huchzermeier and Loch (2000), for example, write that besides variability in budget, schedule and market payoff, also variability in performance and market requirements influences the project execution. According to McDermott (1999), radical product development projects focus on what *could* be, rather than on what already exists. Tatikonda and Rosenthal (2000) describe that for many product development projects, particularly those with uncertainty in technology, the resulting capabilities of the product and the exact means to realize the product are not known with certainty at the start of the development project.

To explain the consequence of technological uncertainty in radical NPD projects we will use the *Work Breakdown Structure* (*WBS*) of the project. The WBS is a deliverable-oriented grouping of project elements that organizes and defines the total scope of the project. Each descending level represents an increasingly detailed definition of a project component (Project Management

Institute, 1996, pp. 171). The WBS is a schematic representation of the disaggregation-integration process based upon which the project manager makes plans to execute the project. The work content of a project is divided into work packages that can be assigned and performed by one of the participating organizational units (Shtub *et al.*, 1994). Thus a work package is the smallest element of a project that is budgetable in terms of money and required number of man-hours. It can be assigned to a single resource. During NPD it is likely that engineers discover errors made by themselves or other engineers (Eisenhardt and Tabrizi, 1995) or discover new problems that have to be solved, which may lead sometimes to a change of the functional specifications. These discoveries result in new work packages that were not foreseen at the start of the process. McDermott (1999) discovered in seven case studies of radical innovation projects that "not all of the events leading to the development of the projects were planned" (pp. 639). Dawson and Dawson (pp. 300, 1998) give three consequences of uncertainty for work packages:

- some work packages may not even be necessary at all, depending on the outcome of one or more previous work packages;
- some work packages may need to be repeated and the number of repetitions might be unpredictable;
- some work packages need to be abandoned before completion, thus a work package may be abandoned before finishing as a result of completion of another work package that was performed in parallel.

These authors did not examine the emergence of new, unforeseen work packages, that according to previous mentioned literature also influences project execution. Dawson and Dawson describe yet another consequence of uncertainty for planned work packages, namely the wide range of possible completion times of work packages.

Technological uncertainty also has an impact on the processing time of work packages. Product development can be seen as a series of problem solving activities. Loehle (1994) argues that scientific discovery is a cumulative series of steps or solved problems all (most) of which must be correct, starting with a "eureka". This eureka or "aha! experience" has been studied in cognitive psychology literature (Reed, 1998; Best, 1995) where it is explained as a flash of insight. According to Reed, the key factor distinguishing insight from other forms of discovery is the suddenness of the solution. Because of this suddenness the time it takes to solve a problem in radical or new product development is highly uncertain and therefore it is difficult to estimate accurately how much time is required for work packages. If the estimate turns out to be wrong and the actual processing time of a work package is longer than planned, more resources are spent on the work package and consequently the development costs of this work package are higher than budgeted. Summarizing, an important characteristic of radical NPD projects is technological uncertainty, that influences project execution in several ways:

- functional product specifications are not fixed at the beginning of the project, but variable and therefore these specifications can be changed during project execution, leading to the addition or deletion of work packages and thus leading to a variable work package structure;
- unplanned or unforeseen work packages can be added to the planned work packages when during the processing of a work package a new problem emerges that requires the definition of a new work package;
- processing times of work packages are very difficult to predict because the solution of work packages is often found very suddenly (aha! experience).

Time-to-market has become another important influence of project execution and an important measure of project execution success (Shtub *et al.*, 1994). To increase market share, it is of great importance that the time-to-market is shorter than that of competitors. Furthermore, a short time-to-market extends the product's sales life and results in higher profit margins (Smith and Reinertsen, 1995; Sander and Brombacher, 2000). Thus, when the importance of a short time-to-market increases, the time constraints of a radical NPD project become stronger. The project due date becomes the project deadline, the time at which the project *has* to be finished, in some cases regardless of whether all functional product specifications are realized. Because of the three characteristics that are described above, it is difficult to estimate accurately the time required for completing the whole project. New unplanned work packages and aha! experiences that 'arrive' late, increase the required project resources (man-hours), compared to the number of resources that were originally planned and if no more resources can be allocated to the project, the planned due date or deadline will be exceeded. But exceeding the deadline is not possible in time-constrained projects. Therefore a time-constrained radical NPD project has different requirements for control than radical NPD projects without time constraints or incremental NPD projects. In the next section these control requirements are discussed.

2.4 Control of radical NPD projects

2.4.1 Mechanistic versus organic approaches

The work packages of a project are processed by resources from different departments or groups within a development organization. The activities that take place by these groups must be coordinated. Coordination requires information processing, and as uncertainty increases, the amount of information processing during the execution of work packages increases also. According to Galbraith (1973) organizations can follow several strategies for processing this amount of information. First, rules, programs and procedures can be used to specify necessary behaviors or methods of working before the work packages are performed. Second, managerial roles can be created in a hierarchy structure. When unplanned events arise that are not foreseen by the rules, programs and procedures, the problem is referred to the manager who then has to make a decision. Third, targeting or goal setting is required when uncertainty increases and managers become overloaded with problems. To prevent overloading, power of decision is given to the organizational levels where the unplanned events originate. In order to ensure appropriate decisions, targets or goals are specified that allow employees to select behaviors or to solve the new problem appropriate to the target. However, when uncertainty increases even more, these three strategies are no longer sufficient and the organization must employ new strategies. Galbraith (1973) introduces four strategies, the first two reduce the need for information processing, and the last two increase the capacity to process information:

1. Creation of slack resources. The number of unplanned events can be reduced by reducing the level of performance of the project, like increasing the scheduled time or the number of resources working on the project, or decreasing the required function specifications (or their quality/ambition level). The longer the scheduled time available, the lower the likelihood of a target being missed, according to Galbraith (pp. 15, see also Thompson, 1967; Lipshitz and Strauss, 1997). Prolonging the scheduled time may cause procrastination. In Chapter 3, Section 3.3.1 we will discuss this further.
2. Creation of self-contained tasks. A functional task design is changed to one in which each group has all the resources it needs to perform its task. In the case of a radical NPD project, this means that a group of engineers is responsible and capable for developing a part of the product. This group must have targets or constraints for their part of the project and it must have the required capacity and capability to be able to realize the targets.

This strategy is also known in production and inventory systems. Bertrand *et al.* (1990) use the concept of self-contained production units that are responsible for realizing production orders in physical production.

3 Investment in vertical information systems. As uncertainty increases information about unplanned events may overload a decision maker. According to Galbraith the information processing capacity of the decision maker can be increased by employing information processing technology. Because nowadays, product development without using computers is not possible, the hardware for information systems is usually already available. Therefore investing in software for decision support or project planning and control techniques is more appropriate here.

4 Creation of lateral relations. Like self-contained tasks or groups, this strategy moves the level of decision making down to where the information exists rather than moving it up the hierarchy. Lateral decision processes are employed without creating self-contained groups, for example by creating a liaison role or a task force.

These strategies suggest that when uncertainty increases, an organization should move from its "mechanistic" approaches to more "organic" approaches. Mechanistic approaches are typified by organizational hierarchies, structured processes that consist of rules, procedures and periodic reviews for project control, and centralized decision-making. Organic approaches are characterized by fluidity and flexibility in the task execution process, rich and frequent communication, decentralized decision-making, and few formal procedures (Tatikonda and Rosenthal, 2000). In this respect we can also refer to Berden, Brombacher and Sander (2000) who write that local (or decentralized) information feedback and feed forward approaches are recommended in unstable, uncertain processes. However, researchers are not convinced that only organic approaches should be used when dealing with uncertainty in radical NPD projects. A balance between firmness and flexibility seems necessary (Tatikonda and Rosenthal, 2000). Ettlie *et al.* (1984) conclude that structural complexity, formality and decentralization reinforce incremental innovation (less uncertain) and structural informality and centralization tends to support radical innovation. The combination of centralization and informality can also be seen as a mix of firmness and flexibility. According to Dewar and Dutton (1986) a combination of centralization and decentralization can be found in empirical research because of a failure by the organization to distinguish between different stages of the development project. In the 'fuzzy front end', a decentralized structure in which lower levels participate in decisions facilitates the circulation of information, and exposes decision makers to new technological innovations. During the project execution phase, a more centralized authority structure facilitates the adoption process by reducing conflict and ambiguity. Sheremata (2000) observes that some researchers concluded that mechanistic and organic approaches must be combined to develop new products successfully. Sheremata concludes that successful development requires structures and processes that increase the quantity and quality of ideas, knowledge and information the organization can access and an integration of these materials into collective action.

Current literature on radical NPD projects seems to support a combination of both mechanistic and organic approaches in controlling radical NPD projects. In the remainder of this chapter we take a closer look at well-known project planning and control techniques. These techniques help to structure the project and its review process and therefore have a more mechanistic nature. We will examine how well these techniques can cope with increasing uncertainty, and an increasing need for organic approaches.

2.4.2 Project planning and control techniques

In this subsection we evaluate well-known project planning and control techniques. Because these techniques are established and have been the subject of much research, we refer to these techniques as *traditional project planning and control techniques*. We will discuss the traditional techniques and describe their characteristics.

According to Meredith and Mantel (1989) a project plan in the broad sense of the word must contain nine elements:

- overview: a short summary of objectives and scope of the project;
- objectives: a more detailed description, including profit and technical goals;
- general approach: a description of the managerial and technical approaches to the work;
- contractual aspects: a complete list of arrangements made with suppliers and customers, project review procedures, technical deliverables and specifications, delivery schedule;
- schedules: an outline of the various schedules and a list of milestone events;
- resources: a description of project budget, cost monitoring and control procedures;
- personnel: a list of expected personnel requirements including necessary

skills or types of training;
- evaluation methods: a procedure to be followed in monitoring, collecting, storing, and evaluating the project;
- potential problems: a list of possible risks or difficulties.

Based on these broad elements of a project plan, a more detailed work breakdown structure can be made. This work breakdown structure can be translated in an activity network or work package structure with resource (and material) requirements. Besides the work package structure, also activity durations and precedence relations are required to schedule the project. Project scheduling deals with the planning of timetables and the establishment of dates during which various resources, such as equipment and personnel, are supposed to perform the activities required to complete the project. Schedules are the cornerstone of the planning and control system (Shtub *et al.* 1994, pp. 301) In a project schedule the precedence constraints of work packages are shown in a network, that graphically portrays relationships between work packages.

Project scheduling techniques evolved in the 1950s when the US government was committed to their Polaris missile program, with thousands of contractors to control within tight time schedules. The increasing complexity of these projects influenced developing methods for better control (Kidd, 1991; Shtub *et al.* 1994). Two planning methods were formulated: the Critical Path Method (CPM) and the Project Evaluation and Review Technique (PERT). The aim of these techniques is to answer the following questions (Shtub *et al.* 1994, pp. 302):
- If each of the tasks goes according to plan, when will the project be completed?
- Which tasks are most critical to ensure the timely completion of the project?
- Which tasks can be delayed, if necessary, without delaying project completion, and by how much?
- More specifically, at what times should each activity begin and end?
- At any given time during the project, what is the amount of resources that should have been spent?
- Is it worthwhile to incur extra costs to accelerate some of the activities?

The authors note that these questions relate to time or the possibility of trading off time for money. This is characteristic for CPM and PERT. These techniques implicitly assume that functional product specifications are defined at the beginning of the project and *must* be realized at any time or cost. Therefore these techniques are most suitable in projects with low technological uncertainty where functional specifications *can* be predicted in advance. The Polaris missile program of the US government had a high technological uncertainty. However, the importance of this program was so high, that cost could be traded off with time. When a new problem was discovered and more resources were required, more resources were indeed allocated to the program. Nowadays this approach does not seem feasible. We have already noted that nowadays time-to-market for new products must be very short. But, to shorten development time, it is difficult to allocate more resources because resources (engineers) are scarce. Thus CPM and PERT are more suitable for projects with little uncertainty. (Note that both techniques assume unlimited resources, thus when resources are limited, these techniques may not even suitable for projects with little uncertainty.) When uncertainty increases, the suitability of CPM and PERT for project planning and control is questioned. This suitability is examined in the next paragraphs.

CPM is a scheduling technique for projects in which hardly any uncertainty exists. When CPM is used, all work packages and their precedence constraints must be known at the beginning of the project and processing times must be estimated accurately. The technique enables the calculation of the critical path, or the longest path of work packages in the network. If the planned finish date of one work package on the critical path is exceeded, the whole project will be late. Work packages not on the critical path have slack and can be delayed temporarily. Two types of slack are possible: free slack and total slack (Shtub *et al.* 1994). Free slack denotes the time that a work package can be delayed without delaying both the start of any succeeding work package and the end of the project. Total slack is the time that the completion of a work package can be delayed without delaying the end of the project, the delay of the work package is compensated by the slack of a succeeding work package. Thus, a delay of a work package that has total slack but no free slack reduces the slack of other work packages in the project. When a project is modeled in a CPM network the six questions written in the previous paragraph can be answered.

PERT is a scheduling technique for projects in which no uncertainty exists with respect to work packages and their precedence constraints, but in which uncertainty may exist with respect to processing times of work packages. PERT assumes that processing times of work packages can be modeled by a Beta distribution. For project scheduling this beta distribution is incorporated by assuming that the expected processing time for each work package can be derived from three different estimates: an optimistic time (a), a most likely time (m) and a pessimistic time (b). With these three estimates, the expected time (TE), and the variance (V) can be calculated by:

$TE = (a + 4m + b)/6$

$V = ((b - a)/6)^2$

The Beta probability distribution function is used in PERT instead of the more common normal distribution, because it is highly flexible in form and can take into account such extremes as where $a = m$ or $b = m$ (Meredith and Mantel, 1989). This Beta function possesses finite lower and upper limits and is thus defined on a finite domain. The estimates of a and b must be so extreme that a very small probability exists that the time actually required will fall outside the range of a to b. When the time estimates are given, the longest path from start to finish can be found in stochastic terms. Thus, when a project is modeled in a PERT-network, this technique also provides the answers to the six questions given in a previous paragraph.

A certain level of uncertainty of work package processing times can be dealt with in PERT, by using the Beta distribution function and the three time estimates. But in radical NPD projects uncertainty is even larger, not only with respect to processing times but also with respect to functional specifications and work package structures. Thus the use of PERT techniques in radical NPD projects can be criticized. In former research this criticism is focused on the following three areas.

Three estimates of work package processing times

Assuming that optimistic, most likely and pessimistic processing time estimates can be made accurately, the use of the Beta distribution function is widely accepted among researchers. However, only few of them question the ability of people to make these estimates accurately. According to Kidd (1991) people usually make optimistic and pessimistic estimates that are not at the 1 and 99 percentiles respectively, but more likely at the 60/70 and 30/40 percentiles (pp. 1977). This behavior will not overly affect the calculation of the expected duration of one work package, but it will reduce the calculated variance of this work package. In turn, it will reduce also the calculated project variance and subsequently it will lead to an over-confident prediction of the project duration. Furthermore, according to Webb (1994) it is human nature to be optimistic when it comes to predicting the future. Optimism may stem from a mixture of subconscious wishful thinking and a genuine misperception of the way things are proceeding (pp. 189). Optimistic behavior when making estimates is not dealt with by CPM and PERT techniques.

Stability of work package structures

In Section 2.3 planned and unplanned work packages were discussed. A change in planned work packages and the emergence of unplanned work packages change the work package structure and consequently the network on which CPM or PERT techniques are based. Dawson and Dawson (1998) examined that the deterministic nature of the traditional project planning and control techniques are not suitable for projects in which there is uncertainty on whether planned work packages will be needed at all (pp. 299). McDermott (1999) described the existence of 'random events' in radical product development projects and noted that the common practices of project planning and control may be detrimental in these radical project environments. Icmeli Tukel and Rom (1998) found that maximizing project quality is nowadays more important than other scheduling objectives examined in literature. In general, we can say that two types of project scheduling exist: resource driven and time driven project scheduling (see for example De Boer, 1998, pp.94). If the project is resource driven, the objective is to minimize maximum lateness over all planned work packages, without exceeding regular capacity constraints. If the project is time driven, the objective is to plan all work packages such that the total amount of non-regular capacity (such as overtime) used is minimal, subject to deadline constraints. However, when project quality is maximized, the work package structure may be *un*stable. For example, when all planned work packages are done before the project due date, new work packages can be added to try to increase project quality. Icmeli Tukel and Rom (1998) suggest that techniques that minimize project makespan or maximize the net present value subject to precedence and resource constraints do not reflect actual projects anymore. A measure of quality as a scheduling objective or constraint is required to reflect practical concerns (pp. 56). Or, in other words, project planning and control techniques should also be able to adapt to a variable work package structure.

Neglecting the non-critical path

CPM and PERT are focused on work packages that are on the critical path and all control mechanisms are used to make sure that these work packages are not delayed, to prevent a delay of the whole project. Work packages on the non-critical paths are neglected. However, according to Kidd (1991) and Shtub (1994) these non-critical paths can also affect the project completion time (see also Van de Wakker, 1993; and Keizers, 2000). Especially when processing times are difficult to estimate and have high variances, actual processing times of non-critical work packages in the project plan may turn out to be longer than actual processing time of critical work packages. Then, it is possible that a non-critical path suddenly becomes longer than the 'old' or current critical path. In other words this non-critical path becomes the new critical path. When the progress of non-critical work packages is ignored and not controlled, control mechanisms (such as allocating more resources to speed up the progress of a work package) will not be used on time and the project is delayed.

Other techniques, such as GERT and VERT were developed to account for more project uncertainty than PERT, but neither of these techniques overcome all of the shortcomings described above. The Graphical Evaluation and Review Technique (GERT) allows various possible probability distributions for processing time estimates, looping back to earlier work packages, and probabilistic branching from a node in the project network. Nevertheless, it remains questionable whether time estimates can be made accurately and the probabilistic routing occurs in a stable work package structure. All known work packages can be planned in advance, it is just not sure whether they all *have* to be performed. GERT does not allow new work packages to be added to the structure during the project. The Venture Evaluation and Review Technique (VERT) allows all work packages to carry three forms of uncertainty: time, cost and performance. Thus instead of PERT, this technique allows the performance of a single work package to be variable (or dependent on the time invested in the work package), but still VERT assumes that all work packages can be planned in advance.

Summarizing, the planning and control techniques, discussed in this section, have the following characteristics:

- all techniques assume that all project work packages are known at the beginning of the project;
- all techniques assume that precedence relations between work packages are known;
- GERT and VERT allow looping back or probabilistic branching in the network of work packages;
- all techniques assume that the work package structure is stable (although the route of a single case through a GERT or VERT network may be stochastic, all possible routes are known);
- all techniques assume that the project deliverables or the functional product specifications are fixed;
- all techniques assume that the project can be controlled by trading off time

  and cost (or cost of resources);
- all techniques assume that work package processing times are either deterministic (CPM) or stochastic (PERT, GERT, VERT) and that these processing times can be estimated accurately by project team members or based on historical data.

Examining these characteristics, two points of interest emerge. First, comparing the characteristics of these techniques and the characteristics of NPD processes described in Section 2.3, it seems that none of these techniques are really suitable for controlling radical NPD processes. Second, these techniques do not appear to be flexible and informal, in other words, these techniques have a mechanistic nature. Shtub *et al.* (1994) note that complex projects change in content over time, and therefore a network constructed in the beginning of a project may be highly inaccurate later. This is one of the reasons why De Boer (1998) developed a hierarchical planning framework in which aggregate data (for example rough estimates of required resources) is used at a higher level at the beginning of the project and detailed information is used at a lower level as soon as it has become available. Traditional project planning and control techniques specify work packages and formalize a network that tend to limit the flexibility that is required to handle changing situations (pp. 355). In other words, these techniques appear to limit the use of organic approaches. It was discussed in Subsection 2.4.1 that when uncertainty increases, both mechanistic and organic approaches are recommended. When mechanistic techniques are applied for NPD process control, we would expect to find an organic counterpart to make it work, if not we expect problems to occur. People involved in radical NPD projects might behave in such a way that organic approaches come to surface, such as flexibility in the work package execution process (adjustment and redefinition of work packages), rich and frequent communication, and decentralized decision-making. To examine whether this is true, the next chapter is focused on human behavior, in particular human behavior that is of importance in performing and controlling NPD processes.

3 Planning and control of radical NPD projects

3.1 Introduction

Uncertainty seems to be the most important characteristic of radical NPD projects. To deal with uncertainty several strategies were introduced in Chapter 2 that either reduce the need for information processing or increase the capacity for information processing. In other words, these strategies decrease uncertainty or increase flexibility. Instead of selecting one strategy, various authors suggest a combination of firmness and flexibility, of mechanistic and organic approaches to deal with uncertainty. In Chapter 2 traditional project planning and control techniques were discussed. It was concluded that these techniques are of a mechanistic nature. In Section 3.2 of this chapter, it is hypothesized what occurs when radical NPD projects are planned and controlled with mechanistic techniques without considering organic approaches. In Section 3.3 the planning paradox is introduced, that causes processes to exceed their due dates. It is suggested that this planning paradox arises when only mechanistic techniques are used for planning and controlling radical NPD processes. Therefore, in Section 3.4 we discuss organic approaches that may be suitable for planning and control of radical NPD projects, and that can be used in combination with the mechanistic project planning and control techniques. Finally, in Section 3.5 we derive conjectures on the suitability of these mechanical techniques for radical NPD projects.

3.2 A mechanistic perspective on planning

3.2.1 Work package structure: well-defined versus ill-defined work packages

From a mechanistic perspective functional specifications of a new product can be defined at the start of the radical NPD project. When functional specifications are fixed a work breakdown structure of the radical NPD project can be made. The functional product specifications are translated and decomposed into separate work packages that all have to be performed in order to develop the new product according to its specifications. When precedence relationships between work packages are determined, the work package structure or project network can be drawn. However, in Chapter 2 it was discussed that when uncertainty in radical NPD projects increases, functional product specifications cannot be defined and fixed at the beginning of the project. Because functional specifications may vary during project execution, the work package structure may also vary. Work packages may be deleted and new work packages may emerge during the project. Furthermore, because of the uncertainty, the content of work packages that are planned to be executed at the end of the project is also uncertain and dependent on the result of preceding work packages. It is likely that these 'downstream' work packages will be defined in more general terms, because there is not enough knowledge to plan them in detail. This is a well-known strategy for reducing uncertainty: decisions about the detailed content of work packages are deferred until additional information becomes available, see for example Galbraith (1973) and Lipshitz and Strauss (1997). Thus, the work package structure will consist of a mixture of well-defined work packages planned to be performed in the short run and ill-defined work packages planned to be performed in the long run. The next step in scheduling the radical NPD project is estimating the processing time of work packages (or the number of man-hours required for processing the work package).

3.2.2 Processing times of work packages: optimistic estimates

Estimates of work package processing times are used by the traditional project planning and control techniques for example to calculate the project completion time. These estimates are made by people having experience and knowledge about these work packages. In this research it is assumed that estimates are made by engineers who are responsible for performing the work packages. These estimates are often subjective and not based on some objective method. Some researchers on project planning have reported that subjective estimates can lead to an over-confident prediction of the project completion time (Kidd, 1991). Webb (1994) agrees that optimism is common when it comes to predicting the future. These arguments are confirmed by cognitive psychology research, revealing that people tend to be optimistic about their productivity. Bazerman *et al.* (1998) studied one form of intrapersonal conflict, namely the result of tension between what people want to do versus what they think they should do. When people are asked what they want, their responses will be emotional, affective, impulsive, and hot headed. When they are asked what they should do, their responses will be rational, cognitive, thoughtful, and cool headed (pp. 229). According to the authors it is more likely that the want option is chosen when uncertainty increases. In radical NPD projects in which work packages are new and uncertain, we expect engineers unconsciously to 'choose' the want option. This means that engineers will be affective and impulsive while estimating processing times. Therefore it is likely that engineers give optimistic estimates that will please the project leader. Kahneman and Tversky (1979) use the term *planning fallacy* for the tendency to hold a confident belief that one’s own project will proceed as planned, even while knowing that the vast majority of similar projects have run late. This implies that people do not use their previous (and negative) experiences when making estimates. Buehler *et al.* (1994) analyzed this planning fallacy in a number of studies with students. Even if the students were instructed to make a pessimistic prediction, the accuracy of the prediction did not improve and fewer than half of the students finished by this pessimistic time. The students in these studies were performing academic tasks. Academic tasks can also be categorized as innovative, problem solving tasks, since they are often uncertain, ill defined and unstable. Thus we may assume that engineers involved in radical NPD projects also suffer from this planning fallacy and make optimistic estimates of their productivity or the processing time required for a work package.

3.2.3 Decrease of uncertainty: time slack in the project plan

When the project leader is aware of the optimism of engineers, he might increase the estimated processing time with a certain number of hours (time slack) to make the estimates more feasible and the project plan in which the estimates are used more realistic. This is a very common approach to cope with uncertainty. Many researchers agree that the longer the scheduled time available, the lower the likelihood of a target being missed (see Chapter 2 of this thesis, and Thompson, 1967; Galbraith, 1973; Lipshitz and Strauss, 1997, Keizers, 2000). It is therefore reasonable to assume that processing times used in the project plan are equal to the optimistic estimates of engineers plus a certain time slack to reduce uncertainty and increase the likelihood that the project plan and its deadline are realized. Based on these calculated processing times, the critical path of the project can be determined. Furthermore, the free slack and total slack can be calculated for all non-critical work packages. The project plan consists of work packages that all have a certain time slack to overcome optimistic estimates and besides this time slack non-critical work packages can also have free and total time slack.

It is thus plausible that a project plan for uncertain radical NPD projects will contain added time slack when it is planned from a mechanistic perspective as was described above. In the next section we examine what happens when engineers start working on work packages that have been planned in that way and try to realize the due dates of the work packages.

3.3 A mechanistic perspective on control

3.3.1 Low time pressure causes low performance level: time slack is lost

Time slack that is put in the project plan postpones the predicted completion times of individual work packages. Now, suppose that the predicted completion times are used as due dates and that engineers have full access to the project plan made by the project leader. The tightness of the due dates causes engineers to experience time pressure. A due date in the distant future will not impose much pressure on an engineer even if the work package workload is high. Buehler *et al.* (1994) examined the influence of an important deadline on the predicted and actual completion times of students performing academic tasks. They found that students with a two-week deadline predicted that they would finish later than those with a one-week deadline. But the deadline also influenced the actual completion times, since students finished their assignment later when their deadline was two weeks rather than one week. An explanation of these results is that a person experiences time pressure when the deadline is close. This time pressure affects the behavior of a person. Seers and Woodruff (1997) discovered this effect when they studied students working on academic tasks. At midpoint, instead of being 50% done, the results accomplished approximate 10% of the work that will eventually be performed by the deadline. People seem to distinguish between early and late, and late calls for increased effort toward work package completion. The authors even found an exponential function across a relatively large number of people, indicating that early progress is relatively stable in contrast to late progress (pp. 184). These results correspond with the results of Gersick (1988) who discovered a *midpoint-transition*. At the midpoint of the allocated calendar time, groups undergo a transition in which the group makes a major jump in progress. Before this midpoint progress is small. Time pressure thus seems to influence the productivity of people. However, Karau and Kelly (1992) write that too much time pressure can also be problematic, for example, because of the stress it causes. But the lack of time pressure, or time abundance can be equally problematic, potentially leading to boredom or dissatisfaction. Karau and Kelly studied the relationship between time pressure and productivity (that was related to solutions found for certain tasks) of groups. One of their findings was that solutions written by groups in both the time abundance and the time scarcity condition were rated lower on originality than solutions written by groups in the optimal-time condition. Thus, there appears to exist an optimal level of time pressure at which productivity is the highest. Below and above this optimal time pressure level, productivity decreases.

This relationship between time pressure and productivity can be explained by the *arousal theory*. The arousal theory assumes that behavior will change as arousal increases (Petri, 1996). Some changes in arousal, as from sleep to alert wakefulness, will result in increased efficiency of performance. Other arousal changes, as from alert wakefulness to extreme emotional arousal, will interfere with efficient responding. This reasoning suggests that an optimal level of arousal exists at which behavior will be most efficient, see Figure 3.1.

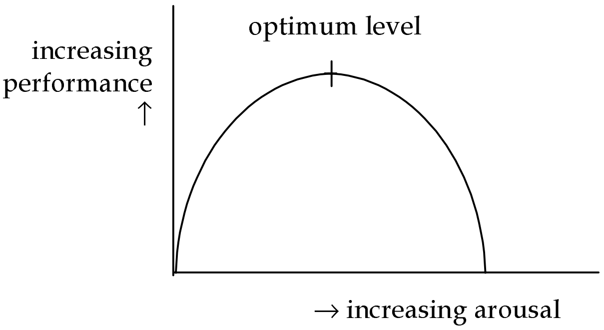


Figure 3.1 inverted U-function relating performance to arousal (Petri, 1996)

Figure 3.1 is also known as the Yerkes-Dodson law, which predicts that the relation between effort and performance will not be monotonic if increased effort induces increased arousal. Trying too hard at a task may cause deterioration in performance, particularly if the task is complex (see Wickens, 1984, pp. 67). Bowers *et al.* (1997) translated this relationship between performance and arousal to a relationship between performance and workload. According to these authors individual performance degrades at low and high levels of workload, whereas a low workload can cause an operator to become complacent or bored, resulting in inattentiveness and reduced performance. The height of the workload can only be measured if the deadline at which the work has to be finished is taken into account. A workload cannot be high if no completion date or deadline is given. Both workload and closeness of the deadline determine the perceived time pressure of an engineer.

In the previous subsection we described that because of added time slacks in the project plan, the deadline is not as near as it would be without these time slacks. In other words, because of added time slacks, the time pressure engineers perceive is low and the corresponding performance or productivity is also low. Assuming that the original optimistic estimates of processing times were based on an optimal productivity level, some kind of self-fulfilling prophecy arises here. The project leader adds a time slack to the optimistic estimate, but because of this time slack, the time pressure declines resulting in a lower productivity level. When the productivity is lower than originally estimated, the engineer will need more time to do the same amount of work, i.e. performing his work package. According to Parkinson's well-known law (1957), which states that the time required for a task is determined by the time available for that task, this 'more time' will be about equal to the time slack given by the project leader. This line of reasoning is shown graphically in Figure 3.2. In this figure a work package consisting of 18 problems has, at a productivity rate of 3 problems per time unit, an estimated processing time of 6 time units. A time slack of 3 time units is given, that results in a lower productivity rate of 2 problems per time unit. In the case of no uncertainty due to disturbances, the work package will be finished after 9 time units. Thus, it is likely that the time slack that is put into the project plan will to some extent be consumed by the lower productivity of engineers.

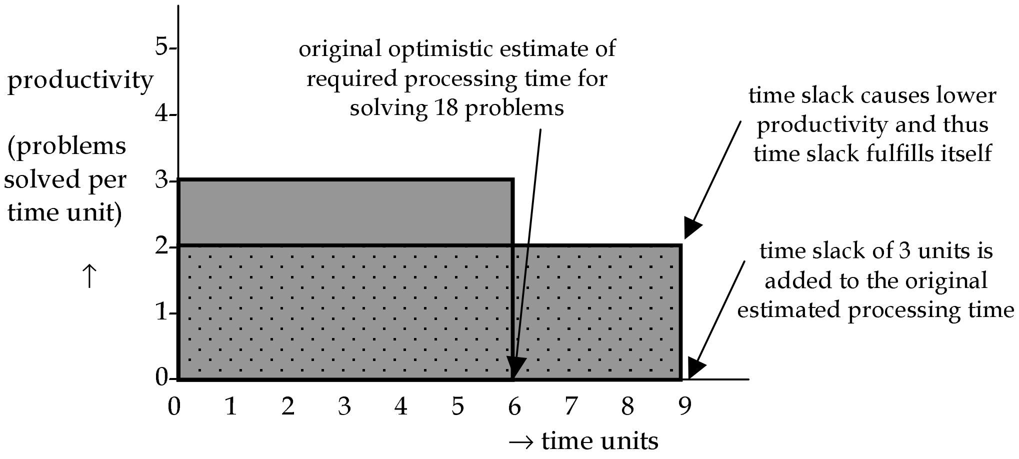


Figure 3.2 self-fulfilling prophecy of time slack

3.3.2 New problems are discovered: the project deadline is exceeded

In Chapter 2 characteristics of project planning and control techniques like PERT and GERT, were described. These techniques assume that functional product specifications can be defined and fixed at the beginning of the project and that the corresponding work package structure remains stable until the project is finished. However, in Chapter 2 characteristics of radical NPD projects were also described. In radical NPD projects functional product specifications are uncertain and variable at the beginning of the project. During project execution, uncertainty decreases and knowledge is gained on the feasibility of functional specifications. These functional specifications may be changed during the project, resulting in a change of planned work packages. Furthermore, even without changing functional specifications the work package structure may change, because of the arrival of uncertain events or discovery of new problems belonging to a single work package. Changing the work package structure influences the project plan and may also influence the project due date. In the previous subsection we discussed the self-fulfilling effect of time slack that is put in the project plan. Because of this time slack, work packages are not finished within the original optimistic estimated processing time. Work packages require more time, and this extra time may be about equal to the time slack, unless an uncertain event or new problem is discovered and suddenly more work has to be done. Because of the uncertainty in radical NPD projects, a work package can increase in size even while an engineer is working on it. If this occurs, the planned due date or deadline of the work package cannot be realized and the deadline is exceeded. Recall Figure 3.2 in the previous subsection. Suppose no time slack was given to the engineer and he worked at a high productivity rate of 3 problems per time unit. At $t = 4$, a big new problem is discovered that has to be solved and that will require 2 time units. Then the engineer will finish the work package at $t = 8$. In the time slack situation, at $t = 4$ the engineer has solved only 8 problems, thus his remaining work load is 10 problems. When he discovers this big new problem, requiring 2 time units, his time pressure increases and he will start working with a higher productivity rate (3 problems per time unit). But even with this higher productivity rate, he will finish the work package at $t = 9.33$. This work package exceeds the planned due date, because it has consumed all time slack and therefore it can affect also the project completion date. A paradoxical situation arises. It is likely that the time slack that is put in the project plan and 'given' to engineers causes a radical NPD project to run late. In the next subsection this planning paradox is discussed further.

3.3.3 The planning paradox

Considering the results of the literature study of traditional planning and control techniques and radical NPD processes, in combination with the knowledge from cognitive psychology research on the effects of perceived time pressure on human productivity a *planning paradox* for radical NPD projects arises. In order to plan projects under uncertainty we need estimates of work package processing times in probabilistic terms. According to traditional production planning wisdom, time slack is needed in the project plan in order to plan the project such that a high reliability is achieved regarding the completion of the project at the due date. Furthermore, people tend to make optimistic estimates of capacity needed for their own work packages; we could compensate for this optimism by putting still more slack in the project plan, assuming that we know to what extent people underestimate the capacity needed. However, all this slack may result in engineers experiencing a low time pressure, assuming that engineers are aware of the time slack that is put in the project plan. Engineers with a low perceived time pressure may be relatively inefficient; their productivity will be low which means that the time slack put into the plan will be consumed by inefficiency and will be lost and no longer be available to cope with the uncertainties in the project. Thus as the deadline of the project approaches, we may expect that most of the slack put initially in the project will be gone and the remaining uncertainty will make the project run late. This is what we call the planning paradox of new product development; the paradox emerges because in new product development the resources, being people, dynamically respond to the planning parameters in such a way that the intended effect of the planning parameters is nullified.

Because traditional project planning and control techniques are developed from a mechanistic perspective, the human behavior of the people involved in the project is not included. Lai (1997) argues that techniques do not manage projects, people do. Tools and techniques can only speed up some mechanical activities of the job; it is the people who carry the projects. Technical skills must be combined with people skills (pp. 175). We propose that both mechanistic and organic approaches are required to deal with highly uncertain situations in which people are involved. Therefore, in the next section, we will describe some organic approaches that may be helpful to overcome the planning paradox. In Chapter 7 we will combine mechanistic and organic approaches into a framework for the planning and control of radical NPD projects.

3.4 Introduction of organic approaches

3.4.1 Short-term firmness and long-term flexibility

Radical NPD projects cannot be planned in detail at the beginning of the project. Because of the technological uncertainty, it is not certain which functional specifications can be realized and it is not certain which work packages are necessary for realizing the functional specifications. Because the work package structure is variable, it is not recommended to make detailed long-term plans covering the entire horizon of the project. Every time the work package structure changes, the detailed project plan must be updated, resulting in an increased planning effort. However, on a short term the work package structure might be more stable. Therefore it is advised to make detailed plans on the short term, the term in which work packages are reasonably certain with respect to content and processing time, and aggregate plans on the long term, the term in which work packages are uncertain with respect to content and processing time, or the term in which work packages cannot even be defined. This is consistent with results of previous research. Thompson (1967), for example, describes the paradox of administration: the dual search for certainty and flexibility. In the short run, administration seeks the reduction of uncertainty, while in the long run, administration strives for flexibility through freedom from commitment (pp. 150). Lipshitz and Strauss (1997) write that predictability can be improved through shortening time-horizons (pp. 153) and Cyert and March (1963) propose that planning is avoided where plans depend on prediction of uncertain future events and planning is emphasized where plans can be made self confirming through some control device (pp. 119). Latham and Seijts (1999) examined the effects of proximal and distal goals on the performance on a moderately complex task. They found that subjects having proximal goals in addition to distal goals perform significantly better than participants in the 'do your best' group (without proximal goals). Proximal goals provide clear markers of progress, and they provide individuals with additional information about performance that is not present when only a distal goal is set.

Thus, we propose that over a short-term horizon work packages should be planned in detail. This will provide engineers with clear proximal goals that support a high performance. Furthermore, because detailed planning on the long term is avoided, not much planning effort is lost and flexibility remains to deal with uncertain future events.

3.4.2 Frequent time-oriented review points

In a radical NPD project many engineers work simultaneously on their work packages. Because of the uncertainty surrounding the project, frequent communication between engineers and the project leader is required to share information on progress and the emergence of new problems. Thompson (1967) defines this as the co-alignment in time and space of both human individuals and streams of institutionalized action (pp. 147). Tripoli (1998) states that in the context of complex work roles, a frequent synchronization of goals should be pursued. Periodically re-evaluating or reviewing is opposed to event-oriented reviewing. When development processes are certain and predictable the timing of review points can be *event-oriented*. Event-oriented review points are often referred to as *milestones*: the end result of one or more work packages (see Turner and Cochrane, 1993). In a certain and predictable process the intermediate and end results of work packages can be clearly identified at the beginning of the process, and therefore the review points (or milestones) can be scheduled accordingly. Turner and Cochrane (1993) found that in projects with well-defined goals the control points or milestones could be synchronized with the achievement of these goals. The hypothesis generated by Gersick (1994) is in conformance with these event-oriented milestones. According to Gersick event-based pacing can be useful when certainty exists and when achieving specific outcomes has more emphasis than meeting deadlines. In uncertain radical NPD processes these event-oriented review points are not possible because it is highly uncertain when the event will occur or even that it will occur (for example the realization of a certain ambition level). Therefore in a radical NPD process it will be more useful to have *time-oriented* review points. Andreasen and Hein (1987, pp. 37) define time-oriented review points or keypoints as 'a checkpoint at which it is of extreme importance to measure the results of the project and the course which it has followed, and perhaps even to stop the project or give it a shot in the arm'. Turner and Cochrane (1993) write that projects with ill-defined goals must have decision points instead of milestones, where the definition of the goals is refined and rebaselined. Gersick (1994) describes that temporal pacing could be useful in uncertain situations in which the path to and specification of the final outcome are at least partly indeterminate, as is the case in radical NPD processes. Therefore, in uncertain radical NPD projects we propose frequent, time-oriented review points that are scheduled at fixed time intervals, and that are used to co-align project team members and their work packages.

3.4.3 Allocation of a few work packages simultaneously

In the literature on project management no information was found about the number of work packages allocated to an individual engineer at the same time. It is usually assumed that an engineer works on one work package at the same time and when that work package is finished he starts working on the next work package. However, in radical NPD processes engineers are involved in problem solving tasks. From studying cognitive psychology literature we observed that during problem solving a certain incubation phase can be distinguished. If attempts fail to solve the problem, it is put away on a conscious level, while at some unconscious level, work proceeds (Best, 1995). When an engineer is in this incubation phase of one work package, he can start with another work package. It may also occur that an engineer has to wait for important information before he can proceed with a certain work package. During this waiting time, he can work on another work package. Wickens (1984) studied similar aspects of tasks and distinguished data limited tasks and resource limited tasks. A task is data limited when adding resources does not improve performance; the performance is limited by the quality of the data. When performance changes with added or depleted resources, the task is resource limited. When a task is data limited, resources may be withdrawn from that task without deteriorating its performance, but these resources can improve the performance of another task. Especially in situations where technology uncertainty is high, we would expect many work packages to be data limited for a certain period (waiting for information or for the completion of other work packages). Therefore we suggest that in many radical NPD projects more work packages should be allocated to engineers at the same time (see also Clark and Wheelwright, 1993).

In many radical NPD projects human resource capacity is scarce and we can consider the engineers as the bottlenecks. By allocating more than one work package to engineers, it is prevented that they cannot work due to, for example an information delay. Thus, instead of giving engineers time slack, engineers are given the opposite: engineers are allocated more work packages at the same time than they can process, to prevent them from 'running idle'.

However, simply allocating many work packages to engineers may cause overload, pressure or stress by engineers. A project leader can prevent this. The project leader gives each work package a planned start and finish date. The start dates of work packages can be planned as early as possible, indicating to engineers that as soon as this start date is passed, it is allowed to start working on the corresponding work packages. The project leader can give priorities to the allocated work packages by planning some finish dates earlier than others. We have discussed already that deadlines influence the behavior of engineers. Seers and Woodruff (1997) acknowledge that the setting of a deadline is a very critical activity, as defining a deadline is the only way to define a midpoint between early progress (low) and late progress (high) (pp. 185). Therefore it is reasonable to assume that engineers prefer to work on a work package that has the nearest deadline (see also Bazerman *et al.*, 1998; and Tripoli, 1998).

3.5 Suitability of traditional project planning and control techniques

Chapters 2 and 3 provide the information to derive five conjectures regarding the suitability of traditional project planning and control techniques for radical NPD processes. These five conjectures are:

1. Traditional techniques, like PERT or GERT, assume that work packages of which the project exists and their interrelationships are known at the start of the process (Meredith and Mantel, 1989; Neumann and Steinhardt, 1979, Adler *et al.*, 1995). However, findings in literature indicate that for radical NPD this is not true. The emergence of an unknown number of unplanned work packages during the project makes the structure of work packages unstable.
2. Traditional project planning and control techniques use probability distributions of processing times on a finite domain (i.e. with lower and upper bounds). But in radical NPD projects some of the problems may turn out to be extremely difficult to solve by engineers, or may turn out to be unsolvable within the time frame given. This level of uncertainty may be far beyond what can be modeled by Beta probability density functions for the duration of work packages (PERT) and what can be modeled by probabilities that work packages will have to be redone (GERT).
3. The processing time of work packages can be estimated most accurately by the engineers working on the radical NPD project, when they are experienced with these kinds of processes. There is no objective estimation method available. Literature studies reveal that engineers are optimistic when they make estimates. As a result the actual processing time will on

average be longer than planned. This is not modeled in traditional project planning and control techniques.

4 The efficiency of engineers may depend on the time pressure they perceive. Thus in order for a project to be carried out efficiently, management must make sure that the optimal level of subjective time pressure is achieved. This is not modeled in traditional project planning and control techniques.

5 In time-constrained radical NPD projects the functional specifications are variable and the time (deadline) is fixed. This is not supported by traditional planning and control techniques. These techniques assume that specifications are fixed and that either the timing or the capacity use of the project is used for controlling purposes.

In the next three chapters of this thesis we report on empirical studies in which we put these conjectures to the test. In Chapter 7 we introduce a framework for the planning and control of radical NPD projects in which both the mechanistic traditional techniques and the organic approaches discussed in this chapter are integrated.

4 ASML, case description

4.1 Introduction

In this chapter a case description is given of the company in which we performed empirical studies. These studies will be discussed in Chapters 5 and 6. In this chapter, in Section 4.2 characteristics of the company are described (mission, customers, net sales, shipments, market share), the manufacturing process of an integrated circuit is outlined and the systems that perform this manufacturing process are briefly discussed (wafer stepper, Step & Scan, TWINSCAN). In Section 4.3 we focus on the new product development processes at the company. Product development phases are distinguished, and the resource allocation process is described. Furthermore the current approach to project planning and control is examined. Finally, in Section 4.4 problems are identified that arise when this current approach to project planning and control is used.

4.2 ASML

4.2.1 The company

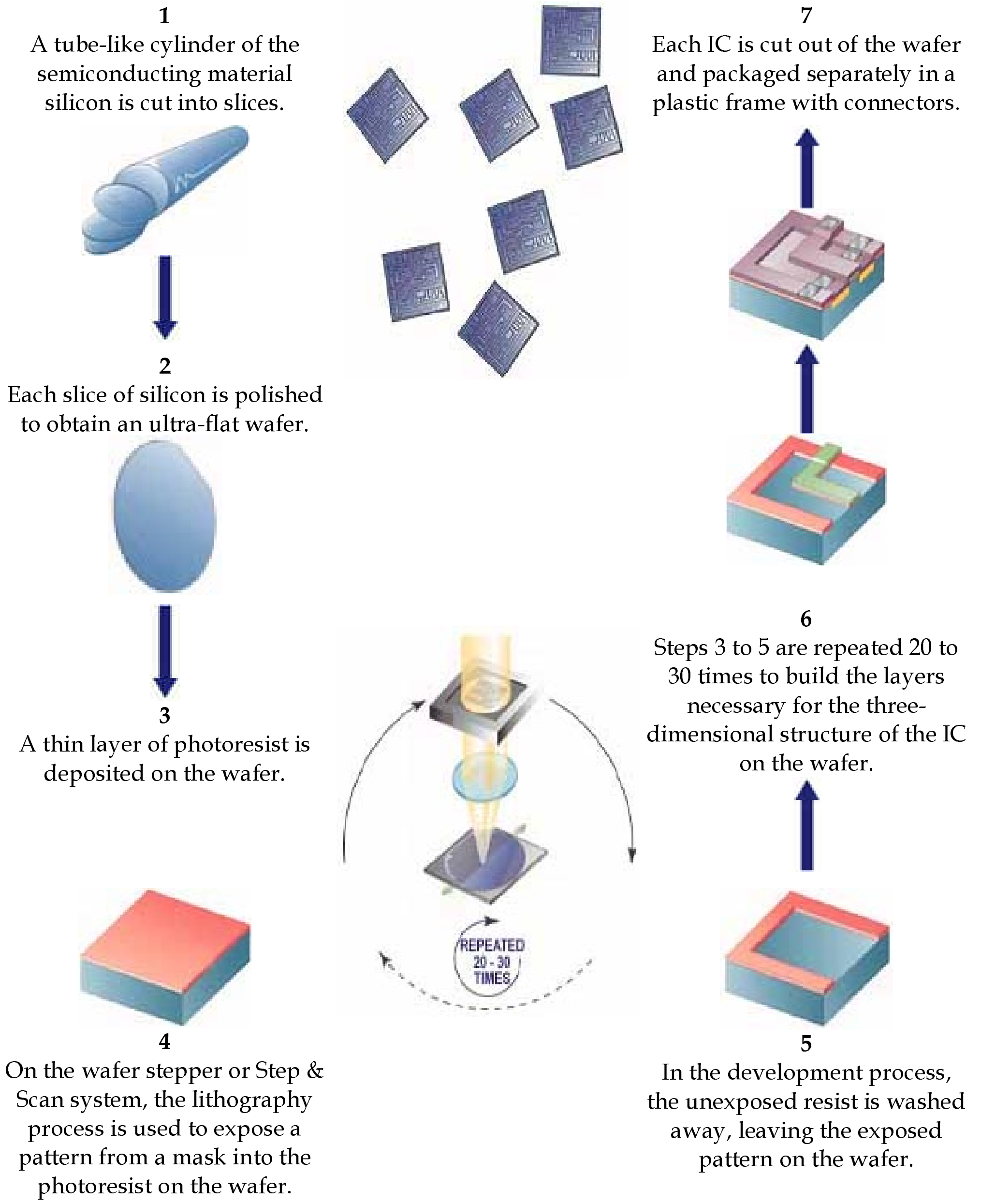
Advanced Semi-conductors Manufacturing Lithography (ASML) was founded in 1984 and develops, produces and services advanced microlithography systems that are used for the production of IC's all over the world. The mission of ASML is: to provide leading edge imaging solutions to continuously improve customers' global competitiveness. Some of the customers of ASML in the global semiconductor industry are: Philips, Samsung, AMD, DEC, Micron, National, SGS-Thomson, TSMC, and Hyundai. In 2000 ASML realized net sales of 2,184 million Euro, compared to net sales of 1,197 million Euro in 1999. In 2000 orders for 464 systems were received, compared to orders for 325 systems in 1999. Unit shipments of ASML's systems were 368 in 2000, of which 264 were Step & Scan systems (or 'scanners') and 104 were Step & Repeat systems (or 'wafersteppers'). These shipments resulted in a backlog of 255 systems at the end of 2000. ASML's total installed base is now more than 1,500 systems. The company is publicly traded on both the Amsterdam Stock Exchange and on the Nasdaq Stock Market. ASML has a manufacturing operation, pilot development, and R&D facility at its global headquarters in Veldhoven, The Netherlands. In addition, applications and training facilities are located in Veldhoven and at its U.S. headquarters in Tempe, Arizona, and in Korea and Taiwan. Regional sales and service facilities are located throughout the world.

In the semi-conductor market, economic pressures and the advent of new electronic applications require more complex ICs (integrated circuits) with higher functionality. To meet this market need and to increase profitability, semi-conductor manufacturers are continually reducing the critical dimensions of their product designs and, consequently, the size of ICs. Microlithography is the enabling technology to realize faster chips, more chips per wafer, and higher yields. ASML works under a time driven control system with a one-year new product release clock speed. Historically, accelerations in technology development and the market demands mentioned above transformed this firm in a very short timespan from a rather small one-product-one-project organization to a highly complex multi-products-multi-project organization, employing 4,377 people. In trying to make its customers competitive in the global semi-conductor market, ASML has gained the largest market share in the Taiwanese and Korean markets as well as among leading-edge IC foundries and DRAM (dynamic random access memory) manufacturers.

4.2.2 The product

ASML supplies lithography systems used for printing IC designs onto the wafers. Integrated circuits and wafer ICs are complex structures of patterns that are printed on silicon, a semi-conducting material. A very thin disk of silicon, commonly known as a ‘wafer’, is used for this. A wafer has a diameter of 20 centimeters but the size of an IC is typically less than one square centimeter, so hundreds of ICs can be printed on one wafer. Each IC is cut out of the wafer and packed separately in a plastic frame with connectors. The name ‘wafer stepper’ is derived from the step and repeat motions that characterize the advanced printing process. Today, the patterns are so fine that it is only possible to print a small area of the wafer at a time. The ‘wafer stage’, one of the many technologically very sophisticated subsystems of a wafer stepper, ‘steps’ the wafer to the new printing position to print the pattern again and again. In this way, the complete surface of the wafer can be used. The printing of these very fine patterns is done by optical means. Each wafer is coated with a thin film of photosensitive material, known as 'resist'. The pattern to be printed is projected onto the wafer using a lens system, quite comparable to the projection of slides onto a projection screen, but almost infinitely more precise. Light induces a chemical change in the resist and the parts that were exposed can now be washed away, leaving a replica of the original pattern on the wafer. The silicon can be treated selectively, for instance by depositing materials or etching them away to build the various components of the circuit.

The optical projection method used in ASML's wafer steppers relies on a reduction lens to project the image of a pattern onto a wafer, a mask that holds the pattern, and an illuminator (light source). The mask, also known as a 'reticle', is held in place above the projection lens by a reticle stage. The projection lens focuses these high-resolution images of the reticle patterns precisely onto a wafer held in place and positioned by the wafer stage. During this critical photomasking process, the wafer stepper aligns the mask image to alignment marks on the wafer. This ensures that the pattern pictured by each mask layer is precisely aligned and 'overlays' the previous mask layers as part of the complete IC manufacturing process. Several thin films have to be stacked on top off each other and each film must be patterned differently. In manufacturing an advanced IC, each wafer must go through the wafer stepper 20 to 30 times. In Figure 4.1 this IC manufacturing process is drawn.



1
A tube-like cylinder of the semiconducting material silicon is cut into slices.

2
Each slice of silicon is polished to obtain an ultra-flat wafer.

3
A thin layer of photoresist is deposited on the wafer.

4
On the wafer stepper or Step & Scan system, the lithography process is used to expose a pattern from a mask into the photoresist on the wafer.

5
In the development process, the unexposed resist is washed away, leaving the exposed pattern on the wafer.

6
Steps 3 to 5 are repeated 20 to 30 times to build the layers necessary for the three-dimensional structure of the IC on the wafer.

7
Each IC is cut out of the wafer and packaged separately in a plastic frame with connectors.

Figure 4.1 the IC manufacturing process

IC production processes at 0.25 micron (which is the thickness of the lines on the chip) and even down to 0.20 micron can be accomplished with conventional wafer steppers. However, at design rules of 0.18 micron and beyond, IC makers are starting to employ Step & Scan technology. Step & Scan technology is more complex and demands a higher level of expertise from the lithography toolmaker in fine mechanics and systems engineering to provide reliable lithography systems for high-volume production applications. For steppers, the full exposure field is illuminated and the reticle image is projected in one flash at each location across the wafer. In Step & Scan systems the illuminator produces a narrow illuminated slit, which projects a part of the image from the reticle onto the wafer. By synchronously moving both the reticle stage and the wafer stage – the scanning action – the whole reticle pattern is imaged onto the wafer. For Step & Scan technology to work, the reticle stage must be perfectly synchronized with the wafer stage during the exposure scan, and the system must be free from any vibrations, which would degrade image contrast and impact pattern overlay. In the second half of 2000, ASML shipped its first TWINSCANTM 300 mm systems. TWINSCAN, the industry's first lithography platform to incorporate two wafer stages, enables exposure of one wafer while another wafer is being aligned. When imaging is completed on the first wafer, the stage holding the next aligned wafer moves into the exposure position. Virtually all overhead time is eliminated. The TWINSCAN 300mm system can pattern twice as many dies per hour as previous systems. Eventually, the industry will reach the physical limits of optical lithography for printing circuit lines and features on semi-conductor devices. Several non-optical technologies are under consideration to succeed optical lithography, and ASML is actively pursuing these candidates on its own and through joint development agreements with partners including Intel, AMD, Motorola, Applied Materials, Lucent Technologies, Samsung and Texas Instruments.

4.3 New Product Development Processes at ASML

4.3.1 Product generation process

The product generation process of ASML is placed within three main sectors: Marketing & Technology, Goodsflow and Customer Support. In general, the product generation process at ASML is divided in six phases.

Phase 0: Formulate product proposals and execute feasibility studies.
Phase 1: Turn study results into system specifications and an integral project plan.
Phase 2: Turn system specifications into detailed design specifications and start sector activities.
Phase 3: Turn detailed design specifications into design realizations and continue sector activities.
Phase 4: Verify design results and finalize sector activities.
Phase 5: Validate results, start operational activities and transfer design to sustaining processes.

All three main sectors of the company are involved in each phase of the product generation process in order to speed up the product generation process and shorten the time-to-market. For example, Goodsflow verifies critical material availability in Phase 0, instead of waiting until the product has been designed completely, and the ordering of critical parts is initiated in Phase 1.

In this research we focus on the development processes performed in the Technology department. The six general phases are translated into six more specific phases for the Technology department. A brief description of the six Technology phases is given below.

Phase 0: *Technical Feasibility*: perform (sub)system studies and product investigations, draft system performance specifications, determine technical risk areas, estimate required resources and competences.
Phase 1: *SPS/SDS Product Plan*: finalize System Performance Specifications, decompose System Design Specifications, set up product specification tree and product plans, allocate project resources and budgets.
Phase 2: *Detailed Product Definition*: define system tests, agree on performance and budget SDS, specify subsystems, modules and mono's (Element Performance Specifications and Element Design Specifications), set up a machine and subsystem integration plan.
Phase 3: *Realization*: build proto type, finalize specifications, architecture and tests, design mono parts, finalize software and (sub)system integration plan.
Phase 4: *Integration*: finalize proto type, perform (sub)system software integration test, define α-integration test plan, accept α-test.
Phase 5: *Validation*: support β-test activities, collect field feedback, solve reported problems, prepare product release, release software and Technical Product Documentation, prepare sustaining processes.

We will not focus on the content of the activities in each phase. Activities are treated as problem solving tasks and we will examine the way in which these problem-solving tasks can be planned and controlled. In the next subsections a description of the current planning and control approach is given.

4.3.2 Claim and allocation process

The Technology department has a matrix organization. This means that engineers can be grouped in two ways. First, engineers are grouped based on their discipline and expertise, where each group has a Group Leader. Second, engineers from different groups are allocated to project teams, in which they are responsible for certain work packages. Each project team has a Project Leader and each team has the objective to realize a certain part of the new machine. The Product Development Manager coordinates all project teams involved in developing a new product.

Three different plans are described here to explain the current project planning process. The Product Development Manager draws up a *master plan* to coordinate the new product development process. It covers the outline of dates and steps required to develop the new product in time. Within this master plan, company milestones are set and main deliverables are identified. The work breakdown of new product development work packages is formulated in different projects. Projects are assigned to a Project Leader and can be regarded as individual projects with their own work packages, budgets and resources. The master plan serves as the controlling level for the *project plans*. Organizing the development projects in this way puts requirements on integration of and reporting on the master plan and project plans. All project plans are based on the outlines set by the master plan. Major milestones at master plan level must be transferred into project plans. On the other hand progress of the different projects must be reported at master plan level. The development Groups (development disciplines) are responsible for supplying the proper resources at the right time, in the right amount and with the right skills. For the long term, disciplines determine resource needs based on the long term Development Program. For short-term resource requirements, communication takes place between Group Leaders and Project Leaders. Project Leaders claim their resources with specific skills or specialized people. Group Leaders allocate the available resources to these project claims. Group Leaders have to optimize the usage of (critical) resources so that these are used as efficient as possible. To simplify this claim and allocation process a *claim and allocation software tool* is in use. Every week Project Leaders can download the most recent group plans and possibly adjust their project plans on Monday and Tuesday. The adjustments are processed on Wednesday. On Thursday and Friday the Group Leaders can adjust their group plans based on the most recent project plans. This process is repeated every week.

4.3.3 Project planning and control

At the beginning of a new product development process at ASML the product specifications are not yet definite and clear. On the one hand the philosophy behind this way of working is saving time. Viewed from a concurrent engineering perspective, the processing time can be shortened by performing processes that used to be sequential, in a parallel or concurrent manner (Clark and Wheelwright, 1993; Krishnan *et al.*, 1997; Prasad, 1996; Smith and Reinertsen, 1995; Stalk and Hout, 1990). This implies that before product specifications are defined completely, the development processes are allowed to start. Then, specification processes and development processes are performed concurrently instead of sequentially, thereby reducing the time-to-market. On the other hand, starting with incomplete and vaguely formulated specifications is inevitable in some processes, because at the beginning of the process there is not sufficient knowledge available to define the specifications more clearly. In this company, sometimes the product specifications are definite just when the product is finished. The projects are planned and controlled with a MS Project software package. Project leaders use Gantt charts and CPM networks for project planning, related to what we denote as 'traditional techniques'. The project team identifies the work packages that have to be performed in order to realize project objectives, and the project team determines the precedence relationships between work packages. Also the processing times of work packages are estimated. These are all 'most likely' estimates (according to the estimators), no explicit pessimistic and optimistic estimates are given, as is required by PERT (see Chapter 2). This information is needed for making a project plan that must be aligned with the master plan of the product. The project plan is used to calculate the planned finish date of the project (is it possible to realize the project due date determined in the master plan?), as a guideline for engineers (when must a certain work package start and/or finish?) and as a standard to compare actual project progress to. A simplified example of a project plan made with MS Project is given in Figure 4.2. The corresponding Gantt chart is also shown in this figure.

ID	Task name	Duration	Done	To Do	Start	Finish	Prede-cessors	Resources
1	wp A	80 hrs	40 hrs	40 hrs	8-1-01	19-1-01		Engr X
2	wp B	40 hrs	0 hrs	0 hrs	22-1-01	25-1-01	1	Engr X
3	wp C	40 hrs	10 hrs	30 hrs	12-1-01	18-1-01	1+4days	Engr Y
4	wp D	20 hrs	0 hrs	0 hrs	19-1-01	23-1-01	3	Engr Y
5	wp E	60 hrs	40 hrs	20 hrs	8-1-01	17-1-01		Engr Z
6	wp F	20 hrs	0 hrs	0 hrs	23-1-01	25-1-01	4;5	Engr Z
7	wp G	60 hrs	0 hrs	0 hrs	29-1-01	7-2-01	2	Engr X
8	wp H	60 hrs	0 hrs	0 hrs	25-1-01	6-2-01	6	Engr Z
9	wp I	60 hrs	0 hrs	0 hrs	7-2-01	16-2-01	7;8	Engr X

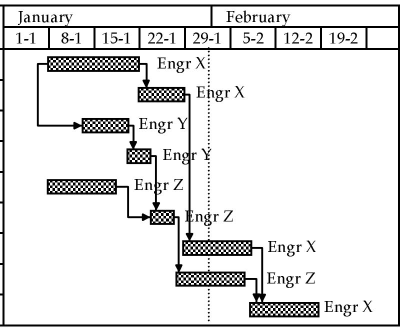


Figure 4.2 a simplified example of a project plan and its Gantt chart

The project plan in Figure 4.2 can also be shown in a PERT diagram when using MS Project. This result is shown in Figure 4.3.

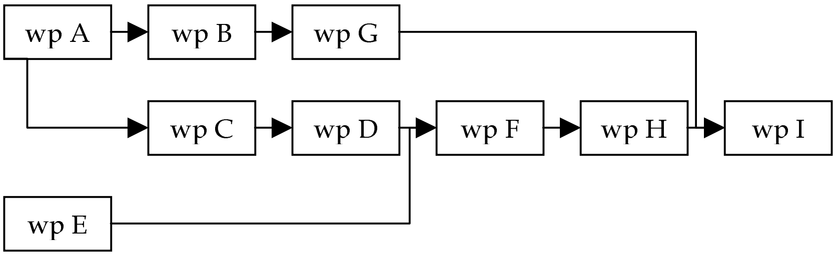


Figure 4.3 PERT network for the project in the example

In the project plan drawn in Figure 4.2 nine work packages have to be performed by three engineers. In this example it is assumed that engineers work eight hours a day, five days a week and they only work on this project. Progress can be measured by comparing the hours engineers have spent ('done') on the work packages with the estimated remaining hours ('to do'). For example, Engineer X has spent 50 hours on work package A, but he estimates that the remaining workload is still 40 hours. This implies that the initial estimated duration of work package A (80 hours) was wrong and should have been 90 hours. Assuming that resources are scarce and no extra resources can be allocated to the project, this expected delay of 10 hours must be included in the project plan. Because work package A is on the critical path, the planned project due date is exceeded and Engineer X will be needed longer than planned. Thus also the group plan of the group Engineer X belongs to must be updated. Project progress is discussed weekly with the project team. Therefore project and group plans can be subject to change every week. Also, when functional specifications are changed, new work packages may emerge or work packages that were planned initially, may not be necessary any more. Thus, changes of functional specifications also require an update of the project plan. In the next section we will describe the operational control problems that arise when projects are planned and controlled in the way described in this subsection.

4.4 Problems with planning and control

The following characteristics of product development projects at ASML are of importance for examining planning and control processes:

1. Product development processes are time-constrained. New product development at ASML is subject to time-constraints that are driven by the market in which ASML operates (accelerations in technology development) and by the time-to-market of competitors.
2. Product development processes are resource-constrained. In product development processes engineers constitute the most important resources. Because it is difficult to employ a sufficient number of academically educated technical engineers, it is plausible to conclude that resources are scarce.
3. Functional product specifications are not fixed but variable at the beginning of a new project, leading to a variable work package structure (or CPM network) and project plan. Furthermore, these functional specifications are used as a control mechanism. For example, if due to the discovery of new work packages the project due date is endangered, it is sometimes possible to change functional specifications in such a way that less work is required and the due date is protected.

Time-constrained radical NPD projects also have these characteristics. In Chapters 2 and 3 we discussed what might happen when radical NPD projects are planned and controlled with mechanistic-oriented techniques like CPM or PERT. It was suggested that work packages will not be finished at their planned due dates, because of the combined influence of optimistic estimates, allocation of time slack, low productivity, and the discovery of new problems (the planning paradox). To examine whether development projects at ASML can be characterized as time-constrained radical NPD projects we performed three small-scale empirical studies. These studies were designed to find characteristics of radical NPD projects that were described in literature (see Chapters 2 and 3). In the first small-scale study we compared estimated and actual due dates of work packages. If actual due dates exceed the estimated due dates, this might suggest that the planning paradox is also present at ASML. In the second small-scale study engineers and project leaders were asked for reasons why planned due dates are exceeded. In the third small-scale study the work package structure of one development project was examined to find some evidence of an unstable structure. If the results of these three small-scale studies indicate that development projects at ASML can be characterized as time-constrained radical NPD projects, a more profound empirical study at ASML of the planning and control processes is justified. In the next chapter we report on the three small-scale empirical studies.

5 Three small-scale empirical studies

5.1 Introduction

In this chapter we report on three small-scale empirical studies that were performed to examine whether development projects in the Technology department of ASML can be characterized as time-constrained radical NPD projects. If so, a more profound in-depth empirical study can be performed in development projects at ASML. Empirical research implies collecting data that refers to well-defined concepts and that are collected under well-controlled conditions. Collecting data about operational processes takes much time and requires the building of trust and the development of a good working relationship between researchers and employees of the firm, which also takes time. Especially building of trust takes much time, and cannot be done in many firms simultaneously. Therefore we decided to perform three empirical studies in three different engineering groups of the development department of one firm. A description of this firm was given in Chapter 4. In order to avoid that the mindset of the engineers (who had to provide the data) was influenced by earlier studies, each empirical study was done only once in a different engineering group. We studied project files of actual projects performed at ASML and interviewed engineers about past performances. We did not study the projects by day-to-day observations as this would require much more time and effort from both the researcher and the firm than was available. In Section 5.2, the first small-scale study is described in which we measured how often the planned flow time of work packages differed from the actual flow time. In Section 5.3 the second small-scale study is discussed in which we examined the causes of differences between planned and actual flow times. In Section 5.4, the third small-scale study is described, in which we studied how often the work packages structure changed during the development project. Conclusions are drawn in 5.5.

5.2 Study of estimated and actual flow times

5.2.1 Methodology

In this first empirical study we have used data available from the project planning department to verify whether work packages of an electronic development project were finished later than estimated or planned a large number of times. Lateness of development processes can be an indication of the arrival of new problems to be solved (more work to be done), and/or of the optimism of engineers regarding the time required to solve the problem. In this phase we do not try to find out why the processes are late, we are only interested in the number of times they are late. If they are late, we are also interested to what extent they are late.

Ten engineers from the electrical engineering group were asked to make estimates of the flow time of all the work packages they had to work on for a period of twenty weeks. The flow time is defined here as the difference between the day on which an engineer starts working on the work package and the day on which an engineer finishes the work package. At the start of each work package they made an estimate of the flow time. When the work package was finished, the actual flow time was noted. All work packages were allocated to the engineers at the same time. Although some precedence relations exist between the work packages, these relationships did not interfere with the measurements because of our definition of flow time (if the start of work package a is delayed because the flow time of the preceding work package *b* is longer than planned, we do not include this delay in the measurement of the flow time of work package *a*). The engineers did not know that their estimates were going to be used in this study. All work packages with the same estimated flow time were taken as one group. Next for each group the actual flow time was compared with the estimated flow time.

5.2.2 Results

In Figures 5.1 to 5.4 the frequency distributions are shown of four extreme situations in which the estimated flow time was both the shortest (Figure 5.1 and 5.2) and the longest (Figure 5.3 and 5.4) estimated flow time that we encountered in this study. Figure 5.1 shows the actual flow time in weeks for all work packages that were estimated to take one week. The same is done in Figures 5.2, 5.3, and 5.4 for the groups of work packages with estimated flow times of two, seven and eight weeks.

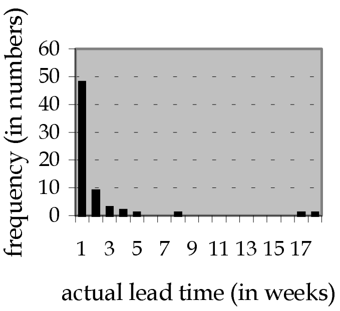


Figure 5.1 frequency distribution of actual flow times for work packages with an estimated flow time of one week (n=66)

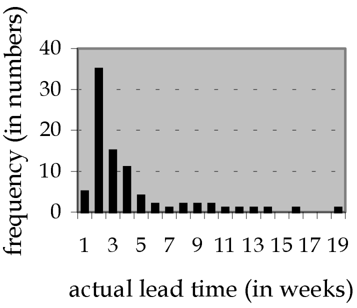


Figure 5.2 frequency distribution of actual flow times for work packages with an estimated flow time of two weeks (n=85)

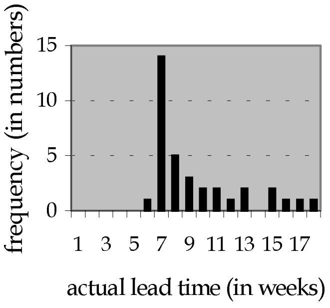


Figure 5.3. frequency distribution of actual flow times for work packages with an estimated flow time of seven weeks (n=35)

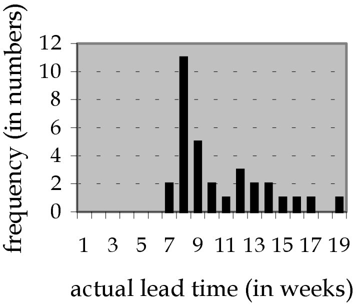


Figure 5.4 frequency distribution of actual flow times for work packages with an estimated flow time of eight weeks (n=32)

These figures show the following:

- Except for the shortest flow time (i.e. the smallest work packages) engineers make the right estimates in less than half of the times. The longer the estimated flow time (the larger the work package), the larger the differences between actual and estimated flow time.
- The frequency distributions all are very skewed. This is consistent with our conjecture that NPD work packages may turn out to contain unforeseen problems, thus requiring much more time than estimated. This is also consistent with the conjecture that engineers are optimistic when they estimate the time required for a work package.
- The actual flow time is almost never shorter than the estimated flow time. This could point to the influence of perceived time pressure on performance. As long as this pressure is low (the deadline is far), the performance of engineers is low. Only when the pressure increases they start working with a higher performance in order to finish right in time. This phenomenon is also known as *Parkinson's Law*: work expands to fill the time available for its completion (Parkinson, 1957). However, we do not know whether this is the reason. The fact that actual flow times are almost never shorter than the estimated flow times could point also to the optimism of engineers when making estimates.

These measurements are consistent with the characteristics of time-constrained radical NPD projects that are planned and controlled with mechanistic techniques, as described in Chapter 3. However, further evidence is needed to verify if development projects at ASML are time-constrained radical NPD projects. Therefore a second and third study have been performed with different groups of engineers and with different research methods. The first study suggested that work packages are often finished too late. In the second study we have investigated the opinion of the engineers and project leaders regarding the reasons for work package lateness. The assumption here is that engineers and project leaders are aware of the underlying determinants of the process and, if questioned, are able and willing to communicate this knowledge.

5.3 Study of reasons for work package lateness

5.3.1 Methodology

In this second small-scale empirical study we let engineers and project leaders think aloud about the development projects at ASML and the reasons for work package lateness. Because we had to be cautious not to bias the participants, we did not expose them to the characteristics of radical NPD projects that were found in the literature, but let them think freely about project lateness. We used the Nominal Group Technique (NGT) to structure small group meetings involving five project leaders and seven engineers of different engineering disciplines (software, physics, mechanics, and electronics).

The NGT allows individual judgments to be effectively pooled and used in situations in which uncertainty or disagreement exists about the nature of a problem or possible solutions (Fox, 1988; Moore, 1987). NGT typically includes four steps:

1 silent generation of ideas in writing (in response to a stimulus question);
2 round-robin recording of ideas (on a flipchart, without discussion);
3 serial discussions of the listed ideas (to understand their meaning);
4 voting to determine the most important ideas.

We gave the participants only one stimulus question and in two meetings of three hours they discussed this question. The stimulus question was: why are project plans not realized? In the first meeting, the twelve participants were split in four groups of three. Within each group the participants first generated their ideas silently and next they combined their ideas on a flipchart. At the end of the meeting each group presented one flipchart with their ideas, i.e. the causes of not realizing project plans. In the second meeting the participants were split again in groups of two (to get a different group composition) and they were asked to cluster the causes if possible and to choose the cluster they found the most important. These clusters should give us a better understanding of the characteristics of NPD and the reasons for lateness.

5.3.2 Results

The participants made ten clusters of causes. In appendix 1 the clusters with all causes are shown. Here we suffice by giving the cluster names and a short description. The number of times a cluster was chosen as 'most important' is shown between brackets (sometimes two clusters were chosen as 'most important' by a group).

A *vague specifications* (3): the functional specifications are the input for product development projects, but because these are formulated vaguely, it is not exactly clear what has to be achieved.

B *varying specifications* (1): specifications that are vague to start with, must change to become more clear and definite. When specifications become clearer, the content or the number of work packages may change, sometimes resulting in more work than expected.

C *specification procedure* (1): a clear procedure on how to define, review, and change specifications is lacking.

D *inaccurately estimating the amount of work (optimism)* (4): engineers explain that they plan too optimistically or that they go along with an unrealistic planning. In reality more time is needed than planned.

E *unplanned activities* (1): engineers are often disturbed during their work by questions from colleagues, telephone calls, discussions, meetings, learning new colleagues the ropes. These unplanned activities diminish the net amount of available time for work packages.

F *balancing between projects* (2): projects are related to each other, if one project is late, the planning of another project is affected (input is not available, so a delay occurs).

G *not working according to the project-approach: planning and tracking* (0): because of the high workload, not enough time is spent on project planning, tracking (progress measurement), and re-planning.

H *non-human resources* (1): a shortage exists of machines or part of machines on which engineers can test their developed solutions.

I *quantity of human resources* (0): people are not focused on one project but are working on more than one project, key project members leave projects too soon due to a shortage of engineers (overload, work pressure).

J *quality of human resources* (0): many new project members are inexperienced or did not have the right company-specific training yet.

Examining these clusters and the ones that were chosen as most important by the participants, we find that some clusters are chosen as most important more often than other clusters (see the numbers between the brackets). The participating engineers and project leaders therefore suggested that the most important reasons why project plans are not realized (or reasons for lateness) are:

- vaguely formulated and varying functional specifications (clusters A, B, C);
- inaccurate estimates of the workload and optimism (clusters D, E).

Clusters F and G are actually consequences of D and E. Because of inaccurate estimates in one project plan, the plans of the dependent projects will also be affected. And, if the flow times are estimated too optimistically, engineers will be allocated more work than they can process. Their workload increases and they will spent less time on planning - and tracking activities. Clusters H, I, and J also give important causes why projects are late. However, these causes are not specific for these development processes, but common for all kinds of projects. Also, participants hardly denoted these clusters as most important.

We may therefore conclude that also the results of this second study suggest that development projects at ASML can be characterized as time-constrained radical NPD projects. To further verify that work package structures are unstable a third study was performed in which the development of work packages belonging to a project is measured over time.

5.4 Study of the stability of the work package structure

5.4.1 Methodology

According to engineers and project leaders participating in the second study, variable or vague specifications are an important reason for not realizing project plans. When specifications are formulated vaguely it is difficult to translate these specifications into a solid work package structure. It is likely that when specifications become more clear or vary, the accompanying work package structure has to change also. Some work packages turn out to be redundant, while other (new) work packages are suddenly introduced. Since this is a characteristic of time-constrained radical NPD projects, the work package structure of such a project will show a variable number of deleted and/or added work packages during the development lead time. To verify whether this occurs also at ASML, we observed a mechanical engineering project at the equipment manufacturing firm during seven months and measured every three weeks the number of deleted and/or added work packages over these three weeks and the total amount of capacity required for the project.

5.4.2 Results

Figures 5.5 and 5.6 show the results of the measurements. Figure 5.5 shows the total number of work packages and the total amount of work (i.e. the total number of required man-hours for the whole project) in the work packages (the lower line). Figure 5.6 shows the number of work packages that was deleted or added every three weeks. Furthermore, the figure shows how many work packages increased or decreased with respect to the amount of capacity required (denoted with > or <). The week in which we measured the data is indicated by the number of weeks before the project deadline.

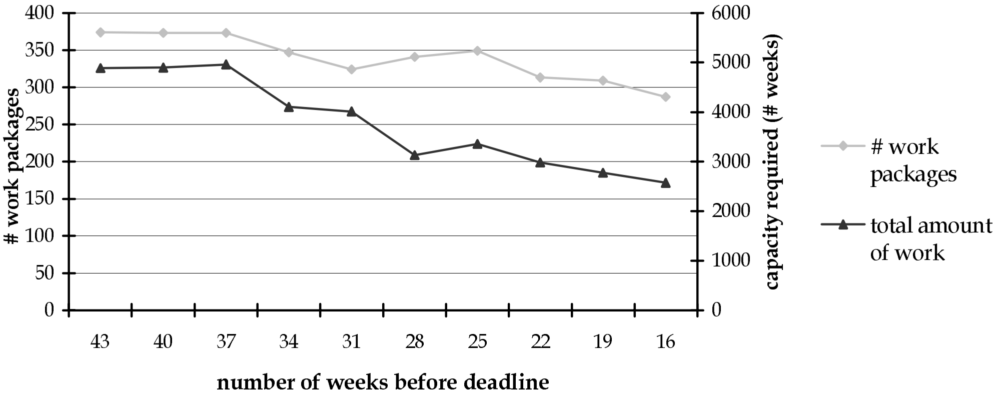


Figure 5.5 number of work packages and amount of work during the project

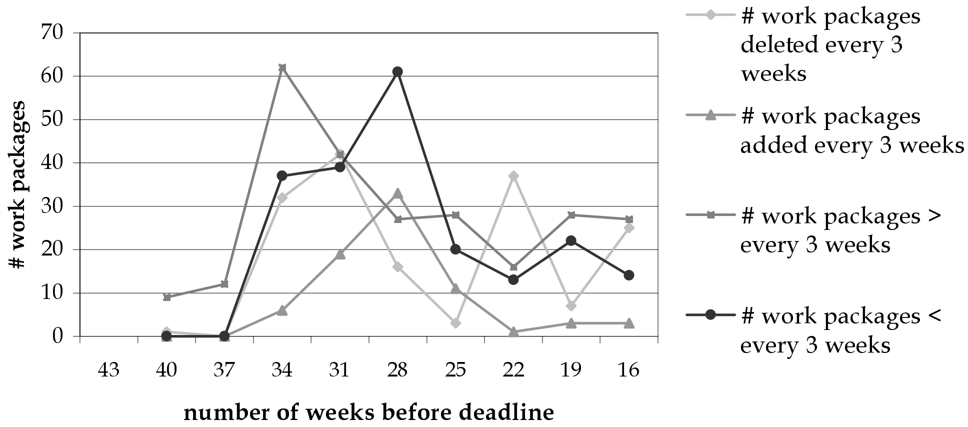


Figure 5.6 number of work packages deleted and added, and number of work packages with increased and decreased required capacity during the project

Figures 5.5 and 5.6 show the varying nature of the work package structure. We see that the structure changes very often due to new work packages added to the project or due to deleted work packages. We also see that the capacity requirements of work packages may change, resulting in a higher or lower workload of the project. This makes it very difficult to realize initial project plans, because the plans must be updated every time the work package structure changes.

Apart from the large fluctuations in the work package structure as shown in Figure 5.6, we see from Figure 5.5 that the total number of work packages and the total amount of capacity required gradually decreases during the project. The explanation of this phenomenon can be found in the way in which the optical equipment manufacturing firm manages its new product releases. The firm has the policy to release new products at a predefined clock speed. Therefore project deadlines are strictly adhered to. As the project deadline approaches, the project managers redefine the functional specifications such that, given the progress already made, and the time and capacity remaining, a maximal functional result is obtained at the deadline. Therefore as the deadline approaches, priorities between functional specifications are determined and gradually certain larger work packages are removed from the project, and sometimes replaced by new smaller work packages. Here, we see that manipulating the functional specifications is part of the controls used by the project management. (Recall that we assume that resources are scarce and no extra resources can be added to the project to speed it up.) The company needs these controls in order to be able to meet the deadline and still realize an acceptable development result. This in part also explains why functional specifications are rather vague and so over-ambitious at the start of the project. Project managers want to have freedom of choice during the development project when responding to the progress made in view of the time remaining until the deadline.

5.5 Discussion of results

The results of these three empirical studies provide important preliminary information in support of our assumption that a planning paradox might exist in radical NPD projects at ASML that are planned and controlled with mechanistic techniques like PERT. In the discussions with the engineers and project leaders the vague and varying functional specifications were denoted as important causes for not realizing initial project plans, but also as characteristics of their work. They did not say that specifications must be *fixed* before the project starts in order to make better plans, but acknowledged that often specifications must be *variable* to make any progress at all, because waiting until they are fixed will take too much time. Variable specifications will result in an unstable structure (number and content) of work packages (third study). Therefore we found support that radical NPD projects seem to be characterized as projects with variable specifications and therefore with unstable work packages structures.

According to the empirical small-scale studies, engineers performing NPD processes seem to be optimistic about their productivity. Optimism was not only denoted explicitly as a cause for not realizing project plans; as a cluster it was also selected four times as the most important reason for delays. This provides us with strong evidence for our third conjecture regarding the optimism of engineers. Because we found some preliminary evidence of the variable work package structure and of the optimistic estimates of engineers, a more profound, in-depth study of engineers involved in product development processes is justified. In this in-depth study we will examine more precisely:
- how many and how often the work package structure changes;
- how optimistic engineers really are in estimating work package processing

times;
- how optimistic project leaders are in making project plans (planning flow times);
- how time pressure and work pressure influence the productivity of engineers.

In the next chapter we first reformulate these questions in hypotheses that can be tested. Then we describe the results of the in-depth empirical study and examine whether the hypotheses can be rejected or not.

6 Empirical study of radical NPD projects

6.1 Introduction

In the previous chapter we reported on three small-scale empirical studies in the development department at ASML. These studies were performed to verify that development projects can be characterized as radical NPD projects, and thus that development projects at ASML are a suitable environment for an in-depth empirical study. The three small-scale empirical studies showed that planned due dates of almost all work packages were exceeded. The nominal group technique was used to encourage both engineers and project leaders to think about reasons for this lateness. They suggested that the most important reasons are optimism and varying functional specifications. Furthermore, preliminary evidence was found of unstable work package structures of projects. The results of these three studies correspond to the characteristics of radical NPD projects identified in Chapters 2 and 3 and justify an in-depth empirical study at ASML, described in this chapter. In this in-depth empirical study, two software NPD projects were closely examined for a period of forty weeks. In Section 6.2 we derive hypotheses based on the literature review that was given in Chapters 2 and 3. In Section 6.3 the methodology is described: the projects are introduced, the data collection method is discussed and the measures are defined. In Section 6.4 results of the in-depth study are discussed, followed by a discussion of these results in Section 6.5.

6.2 Formulation of hypotheses

Stability of work package structures

In Chapter 2 it was identified, based on a literature study, that work package structures of NPD projects are unstable. It was discussed that new work packages can be added to the project network during project execution, caused by a change of functional product specifications or caused by the emergence of an unforeseen problem. Work packages can be deleted from the project network also, due to changing functional specifications. In Chapter 3 we described that because of the uncertainty in radical NPD projects, it is advised to defer decisions about the detailed content of future work packages until additional information becomes available. On the short term work packages can be well defined. This implies that on the short term a detailed project plan can be made because the work package structure is stable on the short term. However, no specific comments are made about the length of this short term. Is this a few weeks or a few months? If the work package structure is stable for a few months, the structure does not have to be changed often and a network technique like CPM or PERT can still be suitable. Therefore, in the empirical study we will explore how severe changes of work package structures are. To explore this, data will be collected to answer the following two questions:

Question 1a: What is the interarrival time of work package structure changes?

Question 1b: When the structure is changed, how many work packages are added or deleted?

Estimating processing times

In Chapter 3 optimistic behavior of people was discussed. It was stated that people tend to be optimistic when they estimate processing times of work packages. Even when past experience indicates otherwise, people have a tendency to hold a confident belief that the new work package can be finished faster. In NPD processes work packages are not likely to be routine and engineers sometimes do not know the content of the work packages precisely

when they have to make estimates. This uncertainty about the content of work packages makes it even more difficult to make accurate estimates. It is likely that for the more complex work packages, it is more difficult to estimate processing times correctly. We assume that more complex work packages will require more time. Therefore we will also examine the differences in optimistic behavior between small and large work packages (or less complex and more complex work packages). Two hypothesis are formulated:
Hypothesis 2a: processing times can be estimated correctly by engineers.
Hypothesis 2b: the longer the estimated processing time (the larger the work package), the larger the estimate error (the larger the optimism).

Planning flow times

In discussions with project leaders and engineers from the development department at ASML information was gathered on how work packages are planned in time. The estimated processing times are the basis for planning the flow times of work packages. Based on the estimated processing times, the project leader calculates the minimum required flow times. This is the number of working days that is required for the work package if the engineer has no other work packages to work on. A normal working day consists of 8 hours, but the ASML rule of thumb states that engineers work on their work packages for at most 80% of the working time. The other 20% of the working time is spent on courses, meetings, illness, etc. Thus, when the project leader calculates the required flow time, he takes into account that an engineer can work 6.4 hours on a work package each day (80% of 8 hours a day). Furthermore, a working week consists of 5 working days; this has to be incorporated in the calculations of the required flow time. As an example, assume that the estimated processing time of a work package is 64 hours. Then, an engineer requires at least 10 working days to process this work package. Thus if he starts on May 1st, he can finish 10 working days later, that is at the end of May 14th. It is important to realize that in the calculation of the required flow time it is assumed that an engineer has only one work package on hand. Therefore we may consider the required flow time as a minimum required flow time. According to project leaders a time slack of 1 day is added to the required flow time. Thus, the flow time that is used in the project plans (i.e. the planned flow time) is this required flow time plus 1 day. To verify whether empirical project data confirm this to be true, we formulate the following hypothesis:
Hypothesis 3: time slack project leaders give to engineers is about 1 day.

If project leaders allow engineers a slack of 1 day, the next question that arises is whether this is slack enough? If engineers have only one work package on hand, this slack might be sufficient, but if engineers have more than one work package on hand simultaneously, it is likely that the workload of engineers (or the required man-hours according to the project plan) turns out to be higher than the available capacity (of 6.4 hours a day) and engineers can be overloaded. Engineers that are overloaded will have difficulties in realizing the planned flow times, and the probability that planned flow times are exceeded increases. Therefore we suggest that if the time slack is only 1 day and if engineers have more than 1 work package on hand simultaneously, then actual flow times will be longer than planned flow times. Two hypotheses are formulated.

Hypothesis 4: engineers have on average about 1 work package on hand simultaneously.

Hypothesis 5: actual flow times are about equal to planned flow times.

Influence of pressure on efficiency

In Chapter 3 the effect of approaching a deadline on the efficiency of people was discussed. The closer the deadline of a work package, the higher the time pressure and the higher the efficiency with which the work package is executed. If engineers have more than one work package on hand simultaneously, they will give a higher priority to the one with the nearest deadline (or earliest due date). The project leader can anticipate this behavior when he assigns due dates to work packages. When a work package must have the highest priority according to the project leader, he can adjust the due date of this work package (setting it earlier than all other work packages an engineer has on hand). In doing so, the project leader controls the *time pressure* of a work package. According to cognitive psychology literature we know that simply increasing the time pressure not always results in higher efficiency, a curvi-linear relationship exist between pressure and efficiency. Therefore we are interested in the question whether or not time pressure of a work package influences the efficiency with which it is executed, and if it does, what the amount of time pressure is that corresponds to the highest level of efficiency. We assume that a work package is executed more efficiently, when the actual processing time is closer to the estimated processing time. We realize that a difference between actual and estimated processing time can also be caused by an estimate error and might not be related to time pressure at all. However, the difference between actual and estimated processing time (the estimate error) is the only variable we can use for comparing time pressure and work

package efficiency. Thus by comparing time pressure and work package efficiency (or estimate error) we do not expect to find strong evidence of a curvi-linear relationship between these two. The comparison can only lead to an indication of this relationship. Furthermore, analyzing the effect of time pressure on work package efficiency is difficult, because time pressure is controlled by the project leader. To protect engineers from overload and stress, the project leader will not let time pressure rise too much. On the other hand, to prevent engineers from running 'idle', the project leader will also make sure that time pressure is not too low. This controlling behavior of the project leader influences our study and gives another reason why we probably will not find strong evidence of this relationship between time pressure and work package efficiency. However, we are interested in the question whether some evidence of this curvi-linear relationship can be found in the empirical study, instead of in a controlled laboratory study. We will therefore perform this analysis and formulate the hypothesis in a general way:
Hypothesis 6: variations in time pressure per work package have no influence on work package efficiency.

If engineers have more than one work package on hand simultaneously, the combination of these work packages induces a certain work pressure on an individual engineer. Time pressure takes only one work package into account. But an engineer will perceive more pressure when he has to finish two work packages at the same due date. To measure this type of pressure, the work pressure of an engineer is introduced here (in which we incorporate all work packages that an engineer has on hand simultaneously, as well as their corresponding due dates) and this work pressure is compared with the engineer's efficiency. Engineer efficiency differs from work package efficiency, because the term engineer efficiency also incorporates all work packages an engineer has on hand simultaneously, instead of work package efficiency that only refers to one work package. Here we must note that also work pressure is controlled by the project leader. To prevent both idle time and overload, the project leader will try to control the work pressure (not too low, not too high), thus we expect that in this study the measurable effect of work pressure on engineer efficiency will be weak.
Hypothesis 7: variations in work pressure per engineer have no influence on engineer efficiency.

6.3 Methodology

6.3.1 Research site: two software radical NPD projects

The two development projects at ASML that are chosen for the empirical study are part of a larger NPD project for developing the first product of a new product family. The projects both consist of software development activities. The reason why these two projects are chosen for our empirical study is twofold. First, software development is an environment in which engineers and project leaders are accustomed already to making detailed project plans, so collecting detailed project data for our empirical study would not disturb their normal planning activities. Second, at ASML software projects (in the realization phase) are considered to be the least uncertain with respect to work package structure and required process and flow times. If we can find support to reject our hypotheses in empirical data from two software projects, it is very likely that the hypotheses would be rejected also in NPD projects that require different disciplines (electronics, physics, mechatronics, optics, etc.). The two projects started in the Fall of 1997, the collection of data started in July 1999 and finished in May 2000. Since our study did not start before July 1999, we could not gather any project data in the period before July 1999. Before we describe the sampling procedure in the next subsection, we will first give some background information on the two software projects and describe how these projects are planned and controlled.

Positioning the two software projects

The two software projects in which we are interested, basic and application software, are part of the larger Software Layout subsystem that has to be developed. Together with other layout projects (e.g. mechanical layout) these projects are grouped for better coordination in a cluster of projects: Usability and Layout, managed by a cluster manager. All clusters of which the new product consists make up the integral NPD project that also has its own planning and management. Application software is a collection of software components in general intended to control and monitor the machine via a user interface. Two groups of software components can be distinguished:

- Lot production related software. With this software a recipe can be defined and run on a machine. A recipe contains all data needed to run a lot. Execution of a lot is controlled by another part of the lot production software. Definition and startup of a lot can be initiated via the user interface of the machine or via a remote control system.

- Software for display of machine status, display of reticle and wafer locations in the machine, user access control, etc.

Basic software is a collection of frames, libraries, etc. that together create the software infrastructure for the new machine that is used by all projects, including the Software Layout subsystem itself. A number of examples is listed here:

- software for communication, error logging, system startup;
- frames and templates for testsoftware, drivers;
- tools and building blocks to create user interfaces;
- tools to install the complete software package on machines.

The Software Layout subsystem has been realized in the time frame from the third quarter (Q3) in 1997 until the first quarter (Q1) in 2000. This time frame was split up into a number of overlapping phases. These phases are described below, with an indication of their time frame (source: Product Management Plan Software Layout S700, by L. Mertens, 26 Feb 1999).

Initial Phase: contribution and review of the system performance and design specification, writing of a project management plan, writing of an initial planning, derivation of budget and resource claim (Q3 1997 – Q4 1997).

Definition Phase: determination of the functionality that is delivered by the Software Layout subsystem, determination of critical software parts (Q4 1997 – Q1 1998).

Realization Phase: realization of all software, starting with critical parts that are identified in the definition phase (Q1 1998 – Q1 2000).

Integration and Test Phase: testing and integration starts when the first deliverables can be integrated, each time a new piece of software is ready it will be integrated with the existing software (Q2 1998 – Q1 2000).

System Integration Phase: integration of subsystems on a number of proto type machines (Q1 1999 – Q1 2000).

Pilot Production Phase (Q3 2000).

Project Release and Transfer to Sustaining Project.

Our empirical research is focused on the realization phase of the two software projects.

Planning and control of the two software projects

During the realization phase the functional specifications of the software project are translated into work packages. Precedence relationships between work packages are defined and work packages are allocated to engineers. Engineers estimate processing times of work packages and discuss their estimates with the project leader. Based on these estimates the project leader makes a project plan and gives each work package a planned start and finish date. These finish dates act as a priority index: the work package with the earliest finish date has the highest priority. Uncertainty in the software NPD projects influences the way in which project leaders plan and control these projects, and consequently it also influences the empirical study.

- Because of the difficulty of estimating the processing times accurately, these estimates are updated every week. For example, when engineers are processing a work package, they may discover that much more time is needed than they initially estimated. The updated estimates are used to evaluate whether the project progress is still conform the original plan or whether the project plan has to be updated (for an example of a project plan, see Chapter 4). When the estimated processing time of a certain work package is adjusted, the project leader can also adjust its planned finish date.
- During project execution new problems may be discovered that require the definition of new work packages. Then these new work packages are added to the work package structure and the project plan is adjusted. Furthermore, it can also occur that a work package is removed from the project plan, because it has become redundant.
- Uncertainty in NPD processes combined with the hard time and resource constraints makes the objective of the process variable. The objective of the software NPD project is to realize a certain desired ambition level of functional specifications. However, it is uncertain whether or not this is possible within the deadline of the project. The more hours engineers can work on work packages the higher the probability that the ambition levels will be realized. Therefore the project leader will prevent engineers from "running idle" by allocating much more work than they can possibly do in a certain period, by setting the same planned start date for several work packages. But the project leader must also prevent engineers from overload or overstrain and therefore the planned finish dates can be delayed sometimes. The workload of an engineer can also be controlled by adjusting the desired ambition levels of work packages.

6.3.2 Sampling procedures

During our empirical study we collected information on the two software projects at the beginning of each week. For each project this information was provided by the engineers and the project leader. Eight basic software engineers and eleven application software engineers estimated the required processing time of each of their work packages as soon as the work package was known or planned. Every week engineers filled in an electronic timesheet to report how much time they worked on each work package and how much time they thought was required to finish the work package. Table 6.1 is an example of such a timesheet.

Table 6.1 example of a timesheet (numbers are in hours)

Name:	**Engineer**		
Project:	**Application Software**		
Period:	**1999-40**		
UID	**Initial Estimate**	**Done**	**To do**
123	18	20	0
456	10	1	10
789	7	5	2

Work packages are given an unique identification number (UID, see column 1 in Table 6.1). When work packages are allocated to engineers, they can fill in their first timesheet. They estimate the number of hours required for solving each work package (initial estimate). In the 'Done' column they can fill in the number of hours they have spent on the work package in a particular period (week). The 'To do' column shows the estimate of the remaining workload of the work package. When this estimate is added to the number in the 'Done' column, a new estimate of the processing time is generated. For example work package 456 has an initial estimated processing time of 10 hours. But in period 1999-40, the estimated processing time is 11 hours (1+10). When 'To do' is equal to 0, the work package is finished. The actual processing time can be calculated by summing the hours an engineer has spent on the work package in each period. The time sheets of engineers are passed to the project leader. Based on the information of the time sheets the project leader can decide to change the project plan. For example, if an engineer estimates in week *t* that the remaining workload of a work package is even more than estimated in week *t* -1, the project leader can decide to postpone the planned finish date of that work package. The weekly updated project plan provided the information for our empirical study. We selected the following items from the project plan:

- date of the project plan;
- UID, the unique identification number of a work package;
- name of the engineer responsible for the work package;
- initial estimate of the processing time;

- number of hours worked on a certain work package in a week ('done');
- estimated number of hours required to finish the work package ('to do');
- planned start date of a work package;
- planned finish date of a work package.

With these items we performed measurements that were required for testing the hypotheses that were defined in the previous section. In the next subsection the measurements required for testing the hypotheses are derived.

6.3.3 Measures

Types of work packages

To explore the instability of work package structures we measured every week the number of work packages each project consisted of. To describe the work package structure, four types of work packages and the interarrival time and size of work package structure changes are discerned:

- *Original work package*: a work package is original if it is known at the beginning of our study and remains part of the project until the end of the project.
- *New work package*: a work package is new if it is not part of the project plan at the beginning of our study but has been defined during the project and appears in the project plan in a certain week, until the end of the project.
- *Deleted work package*: a work package is deleted if it is part of the project plan at the beginning of our study but has been removed from the project plan in a certain week and does not appear again.
- *New-and-deleted work package*: a work package is new-and-deleted if it is not known at the beginning of our study, then in a certain week has been defined and added to the project plan and a few weeks later has been removed again.
- *Interarrival time of work package structure changes*: the interarrival time that can be measured with the available empirical data is the time (in weeks) between two subsequent changes of the work package structure. Because we could monitor the project on a weekly basis, this measure does not reflect the actual interarrival time accurately. For example, if the measured interarrival time is zero weeks, the actual interarrival time is between zero and one week. If the measured interarrival time is two weeks, the actual interarrival time is between two and three weeks.
- *Size of the work package structure change*: the size of the change is the sum of the number of work packages that are added to or deleted from the structure in one week.

By measuring absolute numbers and percentages (of the total number of work packages in the structure) of original, new, deleted, and new-and-deleted work packages per week, and by measuring the interarrival time and the size of work package structure changes, the work package structure is described and the (in)stability of the work package structure can be examined and quantified.

Processing times of work packages

To test whether engineers are optimistic when they estimate processing times of work packages, the following measures are defined:

- *Estimated processing time,* E_p: initial estimate of the processing time of a work package, expressed in hours. The initial estimate is made by an engineer as soon as the work package is allocated to this engineer.
- *Actual processing time,* A_p: actual number of hours the engineer spent to finish a certain work package, expressed in hours. The actual processing time can be measured only when the work package is finished.
- *Estimate error,* $E_p - A_p$: difference in hours between estimated and actual processing time. If the estimate error is negative, the work package needed more hours than initially estimated. This can be the result of optimistic behavior of engineers, but also of inefficient behavior of engineers. If the estimate error is positive, the work package was finished in less hours than initially estimated. This can be the result of either pessimistic and/or efficient behavior of the engineer.

Flow times of work packages

Processing times of work packages are estimated by engineers. Flow times of work packages are planned by project leaders. The flow time of a work package is the time frame during which the work package is on hand. The flow time consists of the processing time plus a certain time slack that is reserved for unforeseen events, waiting for information from other engineers, illness of engineers, etc. To test whether this time slack is realistic and whether flow times are planned accurately, three different flow times of work packages are defined. Furthermore, to perform this test the number of work packages engineers work on simultaneously is required.

- *Required flow time*: R_f: the required flow time is the number of days that are minimally required for processing the work package, based on the estimated processing time. The required flow time is calculated by the following rule of thumb: $R_f = E_p$ / (0.80 * 8 hours).
- *Planned flow time:* P_f: the planned flow time is the initial flow time (in

number of days) of a work package, based on the required flow time and a time slack of 1 day ($P_f = R_f + 1$). The planned flow time is measured just before an engineer starts working on the work package, thus when no progress has been made yet.

- *Actual flow time: A_f*: the actual flow time is the time (in number of days) that elapses between the actual start date and the actual finish date of the work package. The actual start date of a work package is the date at which an engineer starts working on that work package for the first time. The actual finish date of a work package is the date at which an engineers works on that work package for the last time and the estimate of the number of hours required to finish the work package ('to do') is equal to zero.
- *Work packages on hand: wp_{oh}*: the number of work packages engineers have on hand simultaneously in one week. This is the number of work packages of which the planned start date was scheduled in that week or of which the planned start date was already in the past (if the planned start date is in the past, engineers can start working on the work package and the work package is on hand).

Pressure and efficiency

To examine whether a relationship exists between the pressure an engineer perceives and the efficiency with which work packages are processed, two tests are performed. The first test examines each work package separately and ignores all other work packages an engineer might have on hand. Here, the time pressure of a work package is compared with the work package efficiency. The second test examines each engineer separately and includes all work packages he has on hand. Here, the work pressure of an engineer is compared with the engineer efficiency.

- *Time pressure, TP*: the time pressure of a work package (in number of days) is the difference between required flow time and planned flow time ($R_f - P_f$). If this difference is negative, the time pressure is low, because engineers are allowed to spend more days on the work package than required based on the estimated processing time. A positive difference implies high time pressure. For example, when the required flow time is 10 days and the planned flow time is 8 days, the engineer must work very efficiently to finish the work package on time and the time pressure is high.
- *Work package efficiency, Eff_{wp}*: the work package efficiency (in number of hours) is equal to the estimate error ($E_p - A_p$). In the previous subsection it was explained why the estimate error is used for measuring the work package efficiency.

- *Engineer efficiency, Eff_{eng}*: the engineer efficiency (in number of hours per week) in a certain week is calculated by comparing the hours he worked during the week with the progress he made during that week. Suppose an engineer has a workload of 600 hours at the beginning of week 1 (a sum of all hours 'to do'). He spends 30 hours on his work packages in this week. When his workload is 580 hours at the end of the week, the engineer made 20 hours progress and the efficiency in week 1 is –10. But, when the workload is 550 hours at the end of the week, the progress is 50 and the efficiency in this week is +20.
- *Work Pressure, WP*: the work pressure of an engineer (in hours per week) in a certain week is the total number of hours required for all work packages on hand in that week, assuming the (remaining) workload of each work package are spread equally over the flow time of each work package. To explain how the work pressure of an engineer is measured, the following variables are defined:

  $WL_{wp,t}$ =average (remaining) workload per week (in hours) for a work package from t until the planned finish date of the work package
  t =week number, where $t = 1, 2, \ldots$
  wp_i =work package i, where $i = 1, 2, \ldots, n$, and planned start date is in week t
  eng =engineer $A, B, \ldots, S$
  $WP_{eng,t}$ =work pressure for engineer eng in week t, based on all work packages that are planned in week t according to the project plan:

  $$WP_{eng,t} = \sum_{i=1}^{n} WL_{wp_i,t}$$

  An example is given to clarify these measurements.
  It is planned that engineer A has to work on three work packages, wp_1, wp_2, wp_3 in the next 4 weeks. The planned start and finish dates and the average workload for each week is given in Figure 6.1.

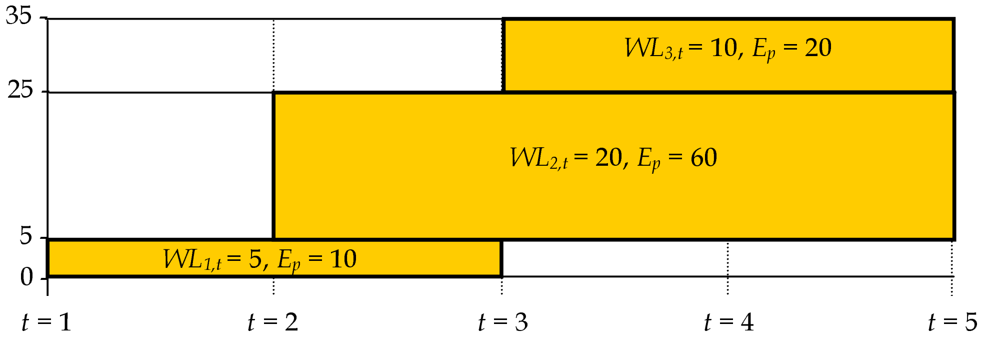


Figure 6.1 plan for engineer A

In week 1 he has to start on wp_1, for 5 hours a week, during two weeks (the estimated processing time, E_p is 10 hours). In week 2 he also has to start on wp_2, for 20 hours a week. According to the plan, wp_2 finishes at the end of week 4. In week 3 a third work package is added to the workload of engineer *A*, with a planned workload of 10 hours a week. If we assume that he works according to plan and the workload is estimated correctly, the following is true:

$WL_{1,1}$ =5 (=10 hours/2 weeks);
$WL_{1,2}$ =5 (=5 hours/1 week);
$WL_{1,3}$ =$WL_{1,4}$ =$WL_{2,1}$ =0;
$WL_{2,2}$ =20 (=60 hours/3 weeks);
$WL_{2,3}$ =20 (=40 hours/2 weeks);
$WL_{2,4}$ =20 (=20 hours/1 week)
$WL_{3,1}$ =$WL_{3,2}$ = 0;
$WL_{3,3}$ =10 (=20 hours/2 weeks);
$WL_{3,4}$ =10 (=10 hours/1 week).

Now we can calculate the work pressure of the engineer per week:

$WP_{eng,1}$ = 5 (only wp_1 is planned to start in t = 1)
$WP_{eng,2}$ = 25 (both wp_1 and wp_2 are planned in t = 2)
$WP_{eng,3}$ = $WP_{eng,4}$ = 30 (both wp_2 and wp_3 are planned in t = 3 and t = 4).

With this measure ($WP_{eng,t}$) we can examine whether workload has an effect on engineer efficiency. We calculate the work pressure of engineers, based on all work packages on hand in the next *three* weeks and relate this number to the engineer efficiency in the first of these three weeks. If the work pressure is high in the first week but low in the next weeks, the engineer will perceive less work pressure than if the work pressure is continuously high in the near future. To examine the influence of work pressure on the engineer efficiency in a week correctly, we measured the work pressure in the next three weeks.

Descriptives of the measures defined in this paragraph are given in Table 6.2, in which *n* denotes the size of the data set.

Table 6.2 descriptives of the parameters

parameter	n	μ	σ^2	σ
E_p	108 wp's	97.6019	7189.7185	84.7922
A_p	108 wp's	116.0093	14097.5420	118.7331
$E_p - A_p$ (= Eff_{wp})	108 wp's	-18.4074	5213.1783	72.2023
P_f	108 wp's	22.2032	412.2733	20.3045
R_f	108 wp's	21.3504	344.0393	18.5483
A_f	108 wp's	70.1698	3283.9823	57.3060
$P_f - A_f$	108 wp's	-47.5394	3287.5347	57.3370
TP (= $R_f - P_f$)	108 wp's	-1.2800	137.8550	11.7412
WP	365 eng-wk	76.2797	1433.0694	37.8559
Eff_{eng}	365 eng-wk	-6.4575	537.6402	23.1871

We checked whether the data gathered from the nineteen engineers are independent by calculating the intraclass correlation between engineers. The intraclass correlation measures the degree of similarity within the same cluster (i.e. all observations belonging to one engineer). The larger the intraclass correlation, the larger the deviation from the assumption of independence between observations. To determine whether or not clustering in the data needs to be taken into account, the design effect needs to be calculated (see Muthén and Satorra, 1995). The design effect is expressed by:

$$\textit{design effect} = 1 + (\textit{average cluster size} - 1) * \textit{intraclass correlation}$$

A design effect greater than 2 indicates that clustering in the data needs to be taken into account. In Table 6.3 the intraclass and design effects are shown for our empirical study. (For example "*eng* x E_p" means that we calculated whether processing time estimates made by different engineers can be considered as independent.) Because the design effects are smaller than or about equal to 2, we can consider the data to be independent.

Table 6.3 intraclass correlations and design effects

	average cluster size	intraclass correlation	design effect
eng x E_p	5.684	0.229	2.073
eng x A_p	5.684	0.146	1.684
eng x R_f	5.684	0.256	2.073
eng x P_f	5.684	0.229	2.199
eng x A_f	5.684	0.082	1.384

6.4 Results

6.4.1 Exploration of the stability of work package structures

NPD projects consist of planned and unplanned work packages. In other words, at the beginning of a project not all work packages are known and during project execution new work packages can be added to the project plan and planned work packages can be removed from the plan, because they are no longer required. In the empirical study we explored how often work package structures change or in other words, what is the interarrival time of work package structure changes? Furthermore, we are interested in the size of structure changes, or the number of work packages that are added or deleted during a structure change. Therefore, we measured weekly the number of work packages (for each type of work package: original, new, deleted, new-and-deleted) the two software projects consisted of. The results of these measurements are shown in Table 6.4 for the application software project and 6.5 for the basic software project. Table 6.4 and 6.5 show for both projects per week, the following:

- week number or date of measurement (column 1);
- total number of work packages of the project per week (column 2);
- number of original work packages per week (column 3);
- number of new work packages per week (column 4);
- number of deleted work packages per week (column 5);
- number of new-and-deleted work packages, i.e. new work packages that are also deleted after a few weeks (column 6);
- interarrival time of work package structure changes (column 7).

The shaded rows in Table 6.4 and 6.5 indicate weeks in which the work package structure is changed (see the next two pages).

Table 6.4 application software

week number	total	original	new	del	new&del	int. arr. time
	in number of work packages per week					in weeks
1	96	96	0	0		
2	102	96	5	0	1	
3	102	96	0	0		
4	105	96	0	0	3	1
5	105	96	0	0		
6	99	89	1	7		1
7	99	89	0	0		
8	99	89	0	0		
9	99	89	0	0		
10	99	89	0	0		
11	118	86	19	3	3	4
12	119	86	1	0		0
13	119	86	0	0		
14	119	86	0	0	1 + -1	1
15	121	86	2	0		0
16	121	86	0	0		
17	127	83	8	3	1	1
18	127	83	0	0		
19	127	83	0	0		
20	143	83	1	0	15	2
21	143	83	0	0		
22	143	83	0	0		
23	143	83	0	0		
24	143	83	0	0		
25	144	83	1	0		4
26	145	83	0	0	1	0
27	144	82	0	1		0
28	144	82	0	0		
29	139	82	1	0	-6	1
30	142	81	17	1	-13	0
31	143	81	1	0		0
32	145	81	2	0		0
33	135	76	0	5	-5	0
34	135	76	0	0		
35	134	75	0	1		1
36	133	74	0	1		0
37	133	74	0	0		
38	133	74	0	0		
39	133	74	0	0		
40	133	74	0	0		
41	133	74	0	0		
42	133	74	0	0		

Table 6.5 basic software

week number	total	original	new	del	new&del	int. arr. time
	in number of work packages per week					in weeks
1	54	54	0	0		
2	55	54	1	0		
3	55	54	0	0		
4	55	54	0	0		
5	55	54	0	0		
6	56	54	0	0	1	3
7	56	54	0	0		
8	56	54	0	0		
9	56	54	0	0		
10	56	54	0	0		
11	60	54	4	0		4
12	59	53	0	1		0
13	57	51	0	2		0
14	57	51	0	0		
15	57	51	0	0		
16	59	46	7	5		2
17	59	46	0	0		
18	59	46	0	0		
19	59	46	0	0		
20	59	46	0	0		
21	61	46	2	0		4
22	61	46	0	0		
23	61	46	0	0		
24	61	46	0	0		
25	61	46	0	0		
26	63	46	2	0		4
27	63	46	0	0		
28	64	46	1	0		1
29	76	41	18	5	-1	0
30	76	41	0	0		
31	76	41	0	0		
32	76	41	0	0		
33	76	41	0	0		
34	77	41	1	0		4
35	77	41	0	0		
36	77	41	0	0		
37	77	41	0	0		
38	89	41	12	0		3
39	89	41	0	0		
40	89	41	0	0		
41	89	41	0	0		
42	89	41	0	0		

Tables 6.4 and 6.5 show the interarrival times of work package structure changes. In Table 6.6 the number of work package structure changes (n) and the corresponding mean interarrival time (μ) of these structure changes are shown. Furthermore, the table shows the mean size of work package structure changes (the mean number of work packages that are added to or removed from the structure when the structure is changed).

Table 6.6 interarrival times and size of work package structure changes

	application software		basic software	
	n	µ	n	µ
interarrival time (in weeks)	19	42/19 = 2.21	12	42/12 = 3.50
size (in # work packages per change)	19	131/19 = 6.89	12	63/12 = 5.25

Table 6.6 shows that the work package structures of the two software projects change about every two or three weeks (the mean interarrival times are 2.21 for the application software project and 3.50 for the basic software project). The mean size of these changes is 6.89 and 5.25 work packages for the application and basic software project. The instability of the work package structures seems to be very high. To show the changes of work package structures over time, we have calculated the relative part of the work package types (original, new, new-and-deleted) in the structure on a weekly basis. Figures 6.2 and 6.3 show these relative parts of each work package type in the structure for both the application and basic software project.

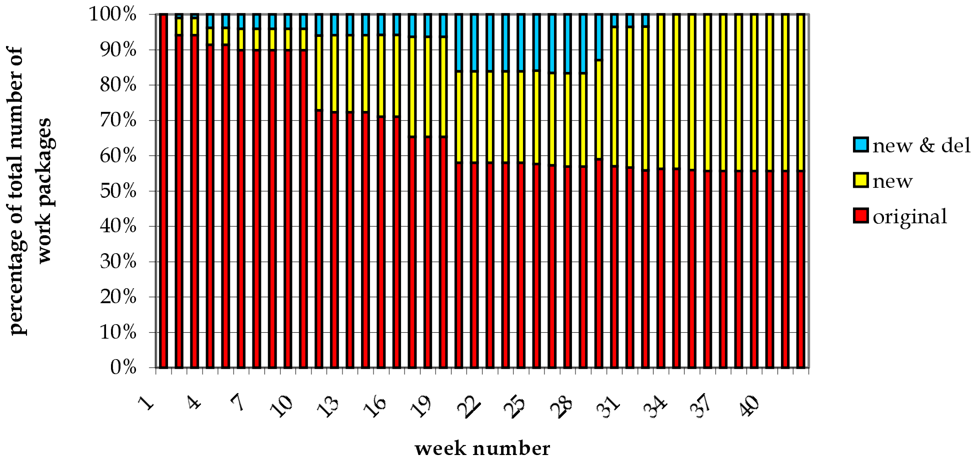


Figure 6.2 structure changes of the application software project

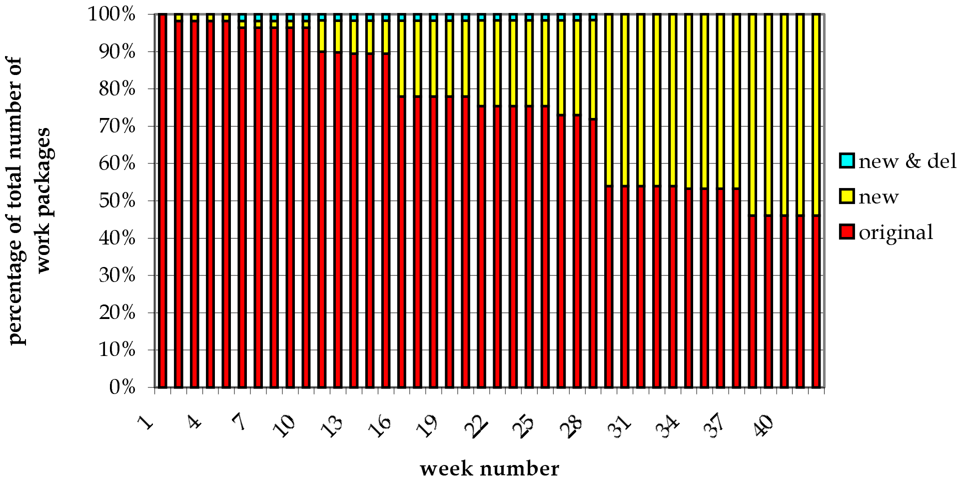


Figure 6.3 structure changes of the basic software project

Tables 6.4, 6.5 and 6.6 and Figures 6.2 and 6.3 show that the work package structures of the two projects change very frequently. Of the original number of 96 application software work packages 22 (22.92%) are deleted during the project. The work package structure at the end of the project consists therefore of 74 original work packages (only 55.64%) and 59 new work packages (44.36%). For the basic software project the results are similar. Of the original number of 54 basic software work packages 13 (24.07%) are deleted. At the end of the project, the work package structure consists of 41 original work packages (only 46.07%) and 48 new work packages (53.93%). Thus, at the end of the project about half of all work packages are new and were not known or planned at the beginning of the project.

The total number of work packages of a project can increase if work packages are split up, for example in order to allocate different parts of the work package to different engineers. Therefore we verified whether the increase in work packages in both the application and basic software projects is the result of a split up of 'old', planned work packages, or the result of the emergence of new work, causing the definition of new work packages. To do so, we measured the processing times of all work packages at the beginning of the data collection (week number 1) and at the end of the project (week number 42). For the application software project we have also measured these processing times in week 26, when the total number of work packages reached its maximum. The estimated processing times made at the beginning can be compared with the sum of the actual processing times at the end of the project. Table 6.7 shows these calculations.

Table 6.7 estimated and actual project processing time (in hours)

project	week	estimated project processing time (in hours)		actual project processing time (in hours)	
application software	week 1 (96 wp)	8.385			
	week 26 (145 wp)	11.656	increase of 39.01%		
	week 42 (133 wp)			12.724	difference is 51.75%
basic software	week 1 (54 wp)	5.621			
	week 42 (89 wp)			7.472	difference is 32.93%

Table 6.7 shows that in week 26 when the application software project counted 145 work packages, the total estimated processing time was increased by 39.01% to a sum of 11656 hours. In the last week of the application software project 133 work packages were measured with a total actual processing time of 12724 hours, that is 51.75% more compared to the original estimate. For the basic software project the difference between total actual (7472 hours) and estimated (5621) processing time is 32.93% of the original estimate. Thus, it seems that new work packages are the result of the emergence of new work.

The results of studying the (in)stability of work package structures suggest that work package structures of these software projects change on average every two or three weeks. The mean size of these changes is between five and seven work packages. Because of all these changes the work package structure consists of about 50% new work packages at the end of the project, while over 20% of all work packages that were planned at the beginning of the project are deleted during project execution. Based on these measures it seems acceptable to say that these software NPD projects have unstable work package structures.

6.4.2 Examination of processing times

In Chapter 2 we conjectured that the processing times of NPD work packages are highly uncertain. The amount of work that has to be done to complete a work package is uncertain and the processing time (or arrival times of aha! experiences) is also uncertain. Because of this uncertainty in NPD we expect engineers to have great difficulty in estimating these processing times accurately. We are interested in the extent to which these estimates are inaccurate. In Figure 6.4 we show both estimated and actual processing times for 108 work packages of both the application and the basic software project.

To test whether the processing times can be estimated correctly by engineers we measured the estimation error (estimated minus actual processing time), its mean and standard deviation and performed a large-sample, one-tailed test for a population mean based on the standard normal z statistic.

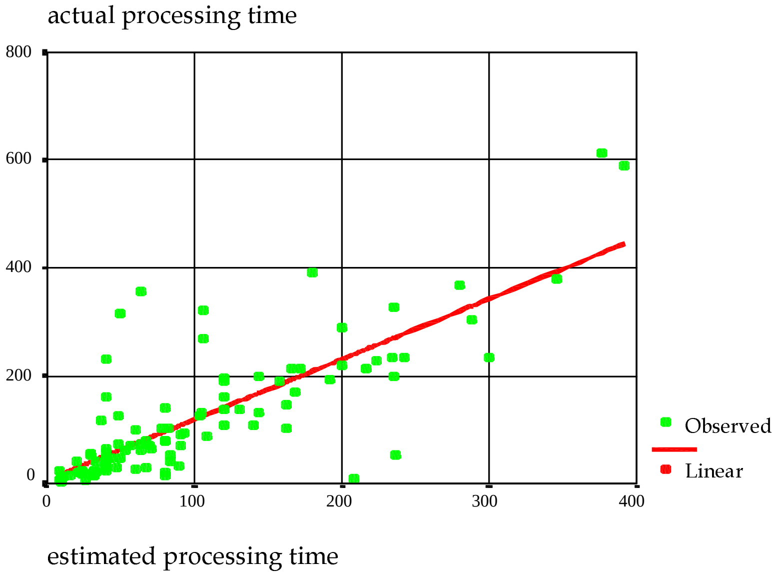


Figure 6.4 comparison of estimated and actual processing time

Hypothesis 2a: processing times can be estimated correctly by engineers.

$$H2a_0 \rightarrow \mu_{E_p - A_p} = 0$$

$$H2a_A \rightarrow \mu_{E_p - A_p} < 0$$

$$\alpha = 0.05$$

μ = -18.4074, where μ is the mean of E_p – A_p

s = 72.2023, where s is the standard deviation of E_p – A_p

n = 108, where n is the number of work packages

$$Z = \frac{\mu - 0}{s/\sqrt{n}} = \frac{-18.4074}{72.2023/\sqrt{108}} = -2.6494$$

$$z_\alpha = z_{0.05} = -1.645$$

Thus, we can reject $H2a_0$, the hypothesis stating that processing times can be accurately estimated by engineers. The results provide evidence for the alternative hypothesis: the estimates are optimistic, because actual processing times are longer than estimated processing times. How much longer (on average) can be shown by fitting a straight line to the data in Figure 6.4. We find:

A_p = 6.9048 + 1.1179 E_p ,

and the coefficient of determination, R^2 = 0.637.

The intercept of this line is about 6.9 hours, which is almost equal to 1 working day. Thus it seems that actual processing times are at least 1 day longer than estimated processing times.

In Section 6.2 we discussed that it is likely that engineers will have more difficulty in making accurate estimates of processing times for larger (and probably more complex) work packages. The linear relationship we found between A_p and E_p suggests that the estimate error becomes larger when the estimated processing time is longer and thus when work packages are larger or more complex. To test whether this is correct, the data set is divided in three equal-sized groups of work packages according to their estimated size (small, medium, large) and test hypothesis 2*b*: *the longer the estimated processing time (the larger the work package), the larger the estimate error (the larger the optimism).* The test statistic that was used for testing hypothesis 2*a* is applied also for testing hypothesis 2*b*, only now the test statistic is applied in each of the three equal-sized groups. The *small* group consists of work packages with an estimated processing time between 8 and 40 hours. The *medium* group consists of work packages with an estimated processing time between 40 and

105 hours. The *large* group consists of work packages with an estimated processing time between 106 and 392 hours. The results are given in Table 6.8.

Table 6.8 estimated versus actual processing times in small, medium and large groups of work packages

	small ($8 \leq wp \leq 40$)	medium ($40 < wp \leq 105$)	large ($106 \leq wp \leq 392$)
n	36	36	36
μ	-0.9167	-23.8056	-30.5000
s	17.8027	79.1313	94.1471
Z	-0.3090	-1.8050	-1.9438

The results in Table 6.8 show that for small work packages the estimate error does not differ significantly from zero (Z = -0.3090). Only for medium-sized and large work packages the estimate error can be considered as negative (Z = -1.8050 and Z = -1.9438 respectively). Thus when engineers estimate processing times that are longer than 40 hours (medium-sized and large work packages), it seems that they are optimistic. There is not sufficient evidence that engineers make inaccurate estimates of processing times when work packages are shorter than or equal to 40 hours (small work packages).

6.4.3 Examination of flow times

In Section 6.2 we discussed that project leaders allow engineers a time slack of about 1 day for each work package. This implies that the planned flow times should be 1 day longer than the required flow times. In the previous subsection we calculated that the actual processing time deviates from the estimated processing time with at least 6.9 hours, and thus approximately one day, especially for the medium-sized and large work packages. In this subsection we test whether empirical evidence can be found to support the assumption that the time slack project leaders put in the planned flow time is 1 day. Therefore we measure the difference between the required and the planned flow time. The required flow time is the time that is minimally required for processing the work package, the planned flow time is the time an engineer is allowed for processing the work package. Therefore the difference between required and planned flow time is a measure for time slack or time pressure. In Figure 6.5 a scatter graph is shown of the 108 work packages that we also used in the previous subsection, but here we measured the required and planned flow times.

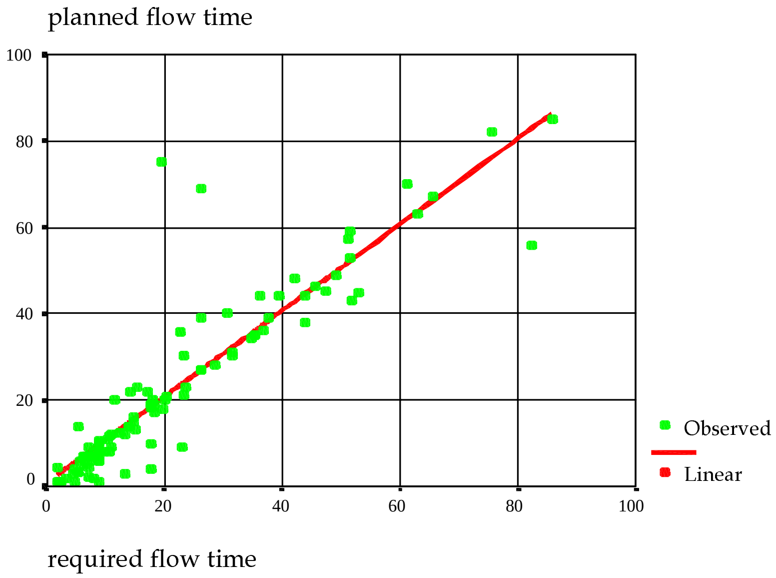


Figure 6.5 comparison of required and planned flow time

If the required flow time is longer than the planned flow time, engineers must finish the work package in less time than possible without working over time, and thus the time pressure is high, or in other words the time slack is small (below the straight line in figure 6.5). If the required flow time is shorter than the planned flow time, the engineers are allowed more time than necessary, and thus the time pressure is low, or the time slack is large (above the straight line in figure 6.5). We calculated the linear regression between required and planned flow time. The slope and the intercept of the straight line can be calculated with the method of least squares (Mendenhall and Sincich, 1988). This results in:

$P_f = 0.9386 + 0.9960 * R_f$, where flow times are measured in days.

The coefficient of determination (R^2) is high: 0,828. Thus, more than 80% of the measured planned flow times are determined by the calculated required flow time. The planned flow time seems to be about 1 day longer than the required flow time (the intercept is 0.9386). A large-sample, one-tailed test for a population mean based on the standard normal z statistic is performed to test whether the time pressure or time slack is 1 day.

Hypothesis 3: time slack project leaders give to engineers is about 1 day.

$$H3_0 \rightarrow \mu_{R_f - P_f} = -1$$

$$H3_A \rightarrow \mu_{R_f - P_f} < -1$$

$$\alpha = 0.05$$

μ = -1,2800, where μ is the mean of $R_f - P_f$,
s = 11,7412, where s is the standard deviation of $R_f - P_f$,
n = 108, where n is the number of work packages.

$$Z = \frac{\mu - \mu_0}{s/\sqrt{n}} = \frac{-1.2800 - -1}{11.7412/\sqrt{108}} = -0.2478$$

$$z_\alpha = z_{0.05} = -1.645$$

Thus we cannot reject $H3_0$. Moreover, since Z is rather small, there seems to be evidence that the slack a project leader gives to engineers per work package is on average 1 day. Because the slope of the regression line is almost equal to 1, we can assume that project leaders give the same slack to all work packages, independent of size or complexity. In Subsection 6.4.2 we found that the actual processing time is on average also about 1 day (or 6.9 hours) longer than the estimated processing time. It seems that the time slack given to engineers, is used to protect against optimistic estimates. However when engineers have more than one work package on hand simultaneously, the workload increases and it is expected that the actual flow times will turn out to be much longer than the planned flow times. (Recall that when planned flow times are calculated, it is assumed that engineers have only one work package on hand.) To verify whether this is true, we will first examine how many work packages engineers have on hand simultaneously (hypothesis 4), and second, we will compare the actual flow times with the planned flow times (hypothesis 5).

Hypothesis 4: engineers have on average about 1 work package on hand simultaneously.

$$H4_0 \rightarrow \mu_{wp_{oh}} = 1$$

$$H4_A \rightarrow \mu_{wp_{oh}} > 1$$

$$\alpha = 0.05$$

$$\mu_{wp_{oh}} = 3.1457$$

$$s_{wp_{oh}} = 1.9947$$

$$n = 494$$

$$Z = \frac{\mu - \mu_0}{s/\sqrt{n}} = \frac{3.1457 - 1}{1.9947/\sqrt{494}} = 23.9095$$

$$z_\alpha = z_{0.05} = 1.645$$

We can reject $H4_0$. The mean number of work packages engineers have on hand simultaneously in these two software projects is about equal to three. (If we test whether $\mu_{wp_{oh}} = 3$, then the test statistic is $Z = 1.6240$.) Thus, we expect actual flow times to be much longer than planned flow times. We will test this by comparing planned flow times with actual flow times of the 108 work packages. In Figure 6.6 a scatter graph is shown of these 108 work packages.

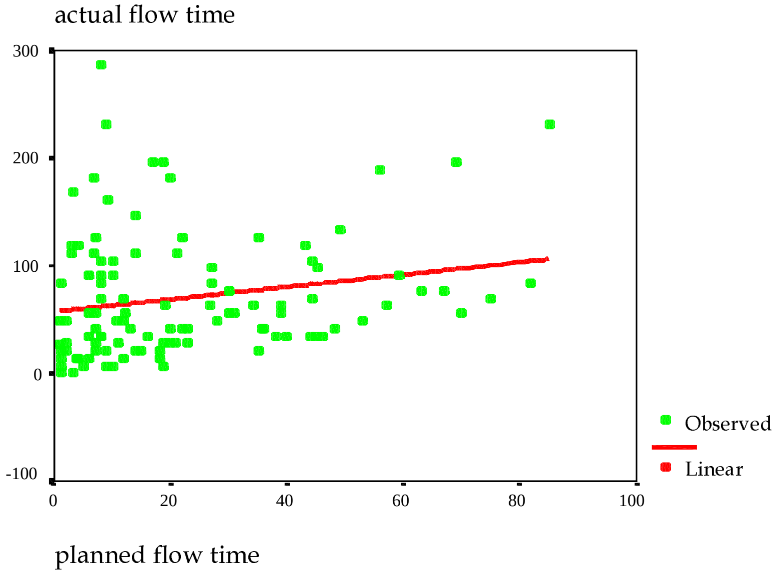


Figure 6.6 comparison of planned and actual flow time

Figure 6.6 shows that there seems to be no relationship between the planned and actual flow time. We have tried to fit a linear, quadratic or cubic model to the data, but none of these models is significant. To test whether the actual flow time deviates significantly from the planned flow time, we test the following hypothesis:

Hypothesis 5: actual flow times are about equal to planned flow times.

$$H5_0 \rightarrow \mu_{P_f - A_f} = 0$$

$$H5_A \rightarrow \mu_{P_f - A_f} < 0$$

$\alpha = 0.05$

μ = -49.9880, where μ is the mean of $P_f - A_f$,

s = 56.3588, where s is the standard deviation of $P_f - A_f$,

n = 108, where n is the number of work packages.

$$Z = \frac{\mu - \mu_0}{s/\sqrt{n}} = \frac{-49.9880 - 0}{56.3588/\sqrt{108}} = -9.2176$$

$$z_\alpha = z_{0.05} = 1.645$$

Thus, we can reject $H5_0$, that the planned flow times are set realistically by the project leader. It seems that project leaders are optimistic like engineers. We have shown that especially for medium-sized and large work packages actual processing times are longer than estimated processing times. Because planned flow times are based on these estimates, it is expected that especially for the medium-sized and large work packages the actual flow times will turn out to be much longer than the planned flow times. Therefore we also test hypothesis 5 in three different groups of work packages. The work packages are grouped according to their estimated size (like we did when we tested hypothesis 2) and the same test ($\mu_{P_f - A_f} = 0$) is performed again. The results are shown in Table 6.9.

Table 6.9 planned versus actual flow times in small, medium and large group of work packages

	small (8 ≤ wp ≤ 40)	medium (40 ≤ wp ≤ 105)	large (106 ≤ wp ≤ 392)
n	36	36	36
μ	-53.9058	-54.3579	-34.3544
s	47.1880	75.2181	43.5622
Z	-6.8542	-4.3360	-4.7318

Table 6.9 shows that regardless of the size of the work package, the actual flow times deviate significantly from the planned flow times. No significant differences were found between these three groups. Thus even when processing times are estimated more accurately (when work packages are small), planned flow times are not realized.

6.4.4 Examination of the influence of pressure on efficiency

According to cognitive psychology literature a curvi-linear relationship exists between work pressure and performance. In this subsection we test whether the time pressure per work package (i.e. the difference between required and planned flow time) influences the efficiency of this work package (i.e. the difference between estimated and processing time). If a curvi-linear relationship exists, work package efficiency will be lower when time pressure is low or high and work package efficiency will be higher when time pressure is medium-sized. However, because the data has been collected in a controlled situation, we will not find many cases in which the time pressure is extremely low or high. A scatter graph of the total data set is shown in Figure 6.7. In this figure a possible curvi-linear line is also shown.

Figure 6.7 shows that indeed the largest part of the data set is centered at the area where time pressure is about zero. It is therefore difficult to fit some curvi-linear relationship between time pressure and work package efficiency. Therefore the total data set is divided into three groups with low, medium-sized and high time pressure and the mean work package efficiencies between these groups are compared. If in an uncontrolled situation a curvi-linear relationship exists, then, in this controlled situation at least differences in work package efficiency between these three groups are expected. The three groups are almost equal in size (two groups with 35 cases and one with 36). The medium-sized group ranges from a time pressure of –0.68 to 0.74, implying that 35 cases were found with a time pressure in this small interval. Descriptives of the three groups are given in Table 6.10. With these descriptives hypothesis 6 is tested.

Table 6.10 work package efficiency in low, medium-sized and high time pressure groups

	low (-13.03 ≤ TP ≤ -0.68)	medium (-0.68 ≤ TP ≤ 0.74)	high (0.75 ≤ TP ≤ 26.47)
n	35	35	36
$\mu_{eff\ wp}$	-26.9714	3.6000	-32.0000
s	72.2949	51.6004	85.2760

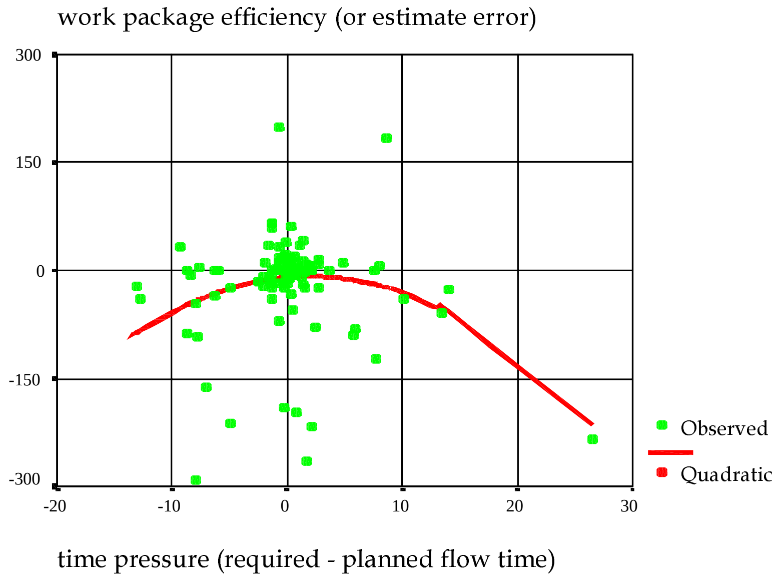


Figure 6.7 comparison of time pressure per work package and work package efficiency

Hypothesis 6: time pressure per work package has no influence on work package efficiency.

$$H6_0 \rightarrow \left(\mu_{eff_{wp}} \middle| TP_{low}\right) = \left(\mu_{eff_{wp}} \middle| TP_{med}\right) = \left(\mu_{eff_{wp}} \middle| TP_{high}\right)$$

$$H6_A \rightarrow \left(\mu_{eff_{wp}} \middle| TP_{low}\right) < \left(\mu_{eff_{wp}} \middle| TP_{med}\right) \quad and \quad \left(\mu_{eff_{wp}} \middle| TP_{high}\right) < \left(\mu_{eff_{wp}} \middle| TP_{med}\right)$$

$\alpha = 0.05$

$Z(TP_{low} \,\&\, TP_{med}) = -2.0363$ and $Z(TP_{high} \,\&\, TP_{med}) = -2.1349$

$z_{0.05} = -1.645$

Thus, we can reject $H6_0$, which states that work package efficiency is not influenced by time pressure. The differences in work package efficiency between low and medium-sized time pressure groups on the one hand and between medium-sized and high time pressure groups on the other hand are significant. When the 'outlier' in the data set (when time pressure is 26.47) is deleted, the mean work package efficiency for the high time pressure group is –26.2000 and the standard deviation is 78.9888. Then, the difference in mean work package efficiency between the medium-sized and high time pressure group is weaker, but remains significant, at $Z = -1.8686$.

In the previous subsection the mean number of work packages engineers have on hand simultaneously was measured. It seems that engineers have on average about three work packages on hand simultaneously. The time pressure of one work package might be low, while the time pressure of the other work packages on hand is high. By looking at all work packages separately, the effect of the combined time pressure of all work packages an engineer has on hand is ignored. Therefore in Subsection 6.3.3 we defined the work pressure of an engineer that incorporates the workload and time pressure of all work packages on hand. We have measured this work pressure of engineers each week, resulting in a data set of 365 engineer-week-combinations. In Figure 6.8 this total data set is shown.

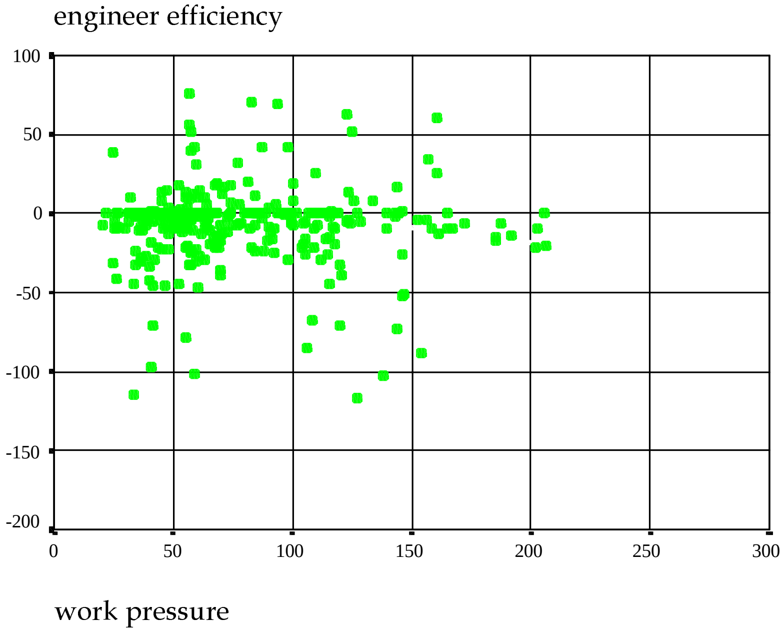


Figure 6.8 comparison of work pressure per engineer and engineer efficiency

Because the measurements are performed in a controlled situation, we see a large part of the data concentrated in a small area on the work pressure axis. In this controlled situation no significant curvi-linear relationship can be found between work pressure and engineer efficiency. Therefore the total data set is divided again in three groups with respectively low, medium-sized and high work pressure to test whether the mean engineer efficiency of these groups differ significantly from each other. In Table 6.10 the descriptives of these three groups is shown. With these descriptives hypothesis 7 is tested.

Table 6.10 engineer efficiency in low, medium-sized and high work pressure groups

	low ($20.04 \leq WP \leq 55.46$)	medium ($55.52 \leq WP \leq 84.28$)	high ($84.36 \leq WP \leq 205.97$)
n	122	122	121
$\mu_{eff\,wp}$	-9.1738	-2.2402	-7.9711
s	20.2713	20.8878	27.3616

Hypothesis 7: work pressure per engineer has no influence on engineer efficiency.

$$H7_0 \rightarrow \left(\mu_{eff_{eng}} \middle| WL_{low}\right) = \left(\mu_{eff_{eng}} \middle| WL_{med}\right) = \left(\mu_{eff_{eng}} \middle| WL_{high}\right)$$

$$H7_A \rightarrow \left(\mu_{eff_{eng}} \middle| WL_{low}\right) < \left(\mu_{eff_{eng}} \middle| WL_{med}\right) \quad and \quad \left(\mu_{eff_{eng}} \middle| WL_{high}\right) < \left(\mu_{eff_{eng}} \middle| WL_{med}\right)$$

$\alpha = 0.05$

$Z(WL_{low}\ \&\ WL_{med}) = -2.6311$ and $Z(WL_{high}\ \&\ WL_{med}) = -1.8341$

$z_{0.05} = -1.645$

Thus, we can reject $H7_0$, that engineer efficiency is not influenced by workload. Instead, engineer efficiency differs significantly between groups with low and medium-sized work pressure and between groups with high and medium-sized work pressure.

In the next section the results of this section are discussed in view of the consequences for the planning and control of these projects.

6.5 Discussion of results

6.5.1 Stability of work package structures

The results of the empirical study clearly show that work package structures change frequently. On average every two or three weeks new work packages must be added or work packages must be deleted from the structure. The projects were examined until they were finished. It was observed that the structure of the application software project stabilizes in week number 37, only 6 weeks before the project is finished. At this time 133 work packages are part of the project. The work package structure of the basic software project stabilizes in week number 39, only 4 weeks before the project is finished. This indicates that projects remain unstable almost until the end of the project. This might be caused by the time constraints and the uncertainty. If the deadline of the project is fixed, the project team will try to solve as many problems as possible before this deadline. This behavior causes the work package structure to change even a few weeks before the deadline. Only then the uncertainty is so low, that a project team can predict accurately which work packages can and will be finished and which not. To deal with unstable work package structures, we propose some consequences for planning and control of these projects:

- Project plans that are made at the beginning of the project, will be obsolete in a few weeks. Therefore it is advised that detailed project plans are made only for the next four to six weeks, depending on the instability of the work package structure (mean interarrival times of work package structure changes). The project plan can be formulated in more aggregate or general terms for the period longer than this four to six weeks timeframe. This will prevent that many hours are spent on planning and replanning the whole project.
- Project status or progress should be measured frequently in order to know whether new work packages must be added to or planned work packages must be deleted from the work package structure. New work packages require more resources or more time and consequently new work packages endanger the realization of the deadline. Since both resources and time are constrained, new work packages will increase time and work pressure. Therefore it is of importance for the project leader to decide which new work packages will be added to the work package structure and which work packages will be deleted.

6.5.2 Estimates of processing times

Hypothesis 2a (processing times can be estimated correctly by engineers) is rejected. However, examining the data in more detail (hypothesis 2b) shows that for small work packages (i.e. smaller than or equal to 40 hours) the hypothesis cannot be rejected. Thus, only for medium-sized and large work packages the actual processing times are significantly longer than the estimated processing times. The relationship between estimated and actual processing times can be expressed quite well by the linear regression line (with $R^2 = 0.637$): $A_p = 6.9048 + 1.1179\ E_p$. This implies that processing times are not as uncertain as we initially conjectured and processing time (or the emergence of aha! experiences) can be predicted by engineers, especially for smaller work packages. The larger the work package, the larger the estimate error will be (the slope of the regression line is 1.1179). Initially we conjectured that the inaccurate processing time estimates (or the emergence of aha! experiences) were the primary source of uncertainty. However, the empirical study results show that work package structure instability is a far more important source of uncertainty. Looking at the consequences of these results for the planning and control of these projects, we suggest the following:

- Estimates of processing times of large work packages will probably turn out to be optimistic. However, the project leader still can decide to use these optimistic estimates for calculating the planned flow time, in order to increase time and work pressure. When engineers discover during the execution of the work package that their estimates were optimistic and the planned finish date of the work package is not delayed, more work will have to be done in the same time, and thereby the time and work pressure is increased.

6.5.3 Planning flow times

Hypothesis 5 was rejected. Actual flow times are significantly longer than planned flow times, for small, medium-sized and large work packages. Thus even when processing times are estimated (almost) correctly, the planned flow times are exceeded. This is not surprising if we take hypothesis 3 and 4 into account. Planned flow times are calculated based on the assumption that engineers have only 1 work package on hand simultaneously. Furthermore, only 1 day of time slack is incorporated in the planned flow times (hypothesis 3). However, the test of hypothesis 4 showed that engineers have on average 3 work packages on hand simultaneously. This implies that planned flow times cannot be realized and thus actual flow times will always be longer than planned. The consequences for planning and control are:

- Allocating more than 1 work package simultaneously to engineers ensures that engineers always have enough work packages to work on. In this way "idle time" and a low perceived work pressure is avoided. Working on 3 work packages each week might be optimal, seen from the efficiency perspective.
- Planned start and finish (or due) dates of work packages have a different nature here than in traditional project planning and control techniques. The planned start dates of work packages indicate when engineers are allowed to start working on the work packages. Thus, the planned start dates indicate when a work package can be considered as 'on hand'. The planned finish dates are used as a 'priority rule'. The engineers consider the work package with the earliest finish date as the one with the highest priority.

Thus, if the project leader delays the planned finish date of a work package, the priority of that work package decreases and furthermore, the time and work pressure will be lower.

- To prevent the work pressure to become too high, thereby overloading the engineers, it is suggested that the finish dates of work packages are controlled. In other words, when the work pressure is too high, the finish date of the work package with the lowest priority is postponed.

6.5.4 Influence of pressure on efficiency

The empirical study on the influence of time or work pressure on respectively work package or engineer efficiency was impaired, because all measurements were taken from a controlled situation (time and work pressure were controlled in such a way that they were not too high or too low). Furthermore the result of studying the influence of work pressure on the engineer efficiency is impaired because the data of all individual engineers are combined in one data set. Probably work pressure has a different influence on the efficiency of different engineers. When data of different engineers are gathered in one data set (see Figure 6.8) it is likely that the curve of the resulting curvi-linear relationship is flattened. These reasons might indicate why in first instance a possibly existing curvi-linear relationship between pressure and efficiency was not found. However, when splitting the data into three groups with low, medium-sized and high time pressure or work pressure, weak but significant efficiency differences were found between the low pressure and medium-sized pressure group and between the medium-sized pressure and high pressure group. Both work package efficiency and engineer efficiency are higher when time pressure and work pressure are medium-sized. Thus for the planning and control of these projects the following is of importance:

- Medium-sized time pressure results in the highest efficiency. In the empirical study medium-sized time pressure was realized when required and planned flow times were about equal. In other words, before the start of the work package, the flow times are planned very tight in order to make sure that the work package efficiency is high. If it turns out that an engineer can work on one work package only (for example because input information for other work packages is not available yet), then this initial time pressure is at the right level.
- As soon as engineers can start working on more than one work package, their work pressure will increase. To prevent work pressure to become too

high, the planned finish date of one or more work packages must be postponed to prevent engineer efficiency to decrease.
- Therefore it is of importance to measure the work pressure of engineers frequently and to adapt the due dates in order to keep the work pressure at a medium-sized level.

The results of the empirical studies and the consequences for the planning and control of radical NPD projects that are derived in this section are used for the development of a new planning and control structure for radical NPD projects. This new planning and control structure is the subject of the next chapter.

7 Requirements for controlling radical NPD projects

7.1 Introduction

In this chapter we will develop requirements for a hierarchical control structure for radical NPD projects. *Control* is defined here as the coordination of activities that are required to develop a new product, that optimizes certain objective functions (for example maximizes quality specifications, minimizes time-to-market or costs). Note that we only discuss the development phase of NPD projects here. We do not consider the manufacturing stage of the new product. Neither do we examine market introduction or service aspects. Developing requirements for the control of radical NPD project is based on both the literature and the empirical study.

The literature study was described in Chapters 2 and 3. In Chapter 3 it was suggested that when mechanistic project planning and control techniques are used for controlling radical NPD projects, also some organic counterpart is required to make the mechanistic techniques work in uncertain environments. In Section 3.4 it was proposed that the following organic approaches might be suitable for the planning and control of radical NPD projects:

- short-term firmness and long-term flexibility (detailed planning on the short term and general planning on the long term);
- frequent time-oriented review points instead of event-oriented review points;
- allocation of more than one work package to each engineer at the same time to prevent 'idle' time.

The empirical study was described in Chapter 6. Hypotheses about characteristics of radical NPD projects were tested. In Section 6.5 the test results were discussed and based upon the consequences for the planning and control of radical NPD projects, the following guidelines were proposed:

- detailed, short-term project plans and general, long-term project plans;
- frequent measurement of project status or progress;
- use of optimistic processing time estimates in the project plan to increase time and work pressure;
- allocation of more than one work package to each engineer at the same time;
- use of planned finish dates of work packages as a priority rule and as a mechanism for controlling work pressure;
- control of work pressure to prevent inefficiency;
- frequent measurement of work pressure.

Besides the literature and the empirical study, the development of requirements for controlling radical NPD projects is inspired by the work of Bertrand, Wortmann, and Wijngaard (1990), who emphasize that a proper control structure should be established first, before techniques and methods can be discussed. They state that a control structure should identify main decision functions, their goals, their (inter)relationships, and organizational positions to which these decision functions should be assigned. For production and inventory systems this is a well-known approach, see for example Meal (1984). However, literature on radical NPD project control usually deals with project phasing (Ulrich and Eppinger, 1995; Andreasen and Hein, 1987; Wheelwright and Clark, 1995) or resource-constrained project scheduling (Özdamar and Ulusoy, 1995; Lawrence and Morton, 1993; Davis and Patterson, 1975). A control structure, similar to the one described by Bertrand *et al.* for manufacturing processes, can be established also for radical

NPD projects. In this chapter we will describe requirements for such a control structure. Though this structure provides guidelines for both the planning and control of radical NPD projects, we will use, in short, the term 'control structure' here.

Bertrand *et al.* define the *production control problem* as follows (pp. 19): Given certain consistent objectives regarding customer delivery performance and manufacturing costs, how should we:

- accept customer orders;
- place production and procurement orders;
- vary the capacity;
- allocate available capacity to manufacturing steps.

For radical NPD projects, the control problem can be defined as: Given certain minimal objectives regarding functional specifications of the new product, given the deadline at which the new product has to be available, and given the resource capacity, how should we:

- determine desired ambition levels of functional specifications of the product;
- define work packages for engineers;
- vary ambition levels of work packages and/or work package structures;
- allocate work packages to engineers.

To solve this radical NPD project control problem a hierarchical control structure needs to be established. A hierarchical approach is required, because that enables the control problem to be decomposed into a number of hierarchically ordered sub-problems and because that reflects the nature of real-life radical NPD projects. The phrase 'hierarchical' is based upon the work of Schneeweiss (1999), who uses it to characterize a situation of at least two levels which exhibit some asymetrical relationship as to their decision rights or their information status, or simply by the fact that one level is making its decision earlier than the other (pp. 9). In time-constrained radical NPD projects, three levels may be distinguished: the product control level, the project control level, and the work package control level. The decisions made in these levels are dependent and therefore they must be coordinated. Schneeweiss distinguishes three stages of interdependence: anticipation, instruction and reaction. The first step is anticipation. The higher level anticipates the characteristics of the lower level in finding a feasible solution. Anticipation can be regarded as feed-forward or as bottom-up influence. The second step is instruction. The higher level makes a decision that influences the lower level and that is based on the anticipation. Instruction can be regarded as top-down influence. The third step is reaction. The lower level responds to the instruction from the higher level. Reaction can be regarded as feedback or as bottom-up influence after the instruction has been given. These interdependencies between product, project and work package control level will be identified in this chapter.

The three control levels correspond to the decomposition in subproblems of the overall problem of developing a new product. Subproblems are allocated to project teams. Project teams solve these subproblems by further decomposing them in work packages that are executed by engineers. This decomposition, corresponding with the three control levels, and the responsible persons are shown in Figure 7.1. Our empirical research, described in Chapter 5, was focused on the project and work package control level. However, during this empirical research information was gained on control processes at the product control level, and based on a literature study we can give recommendations for the control structure on the product control level as well (this is described in the next section).

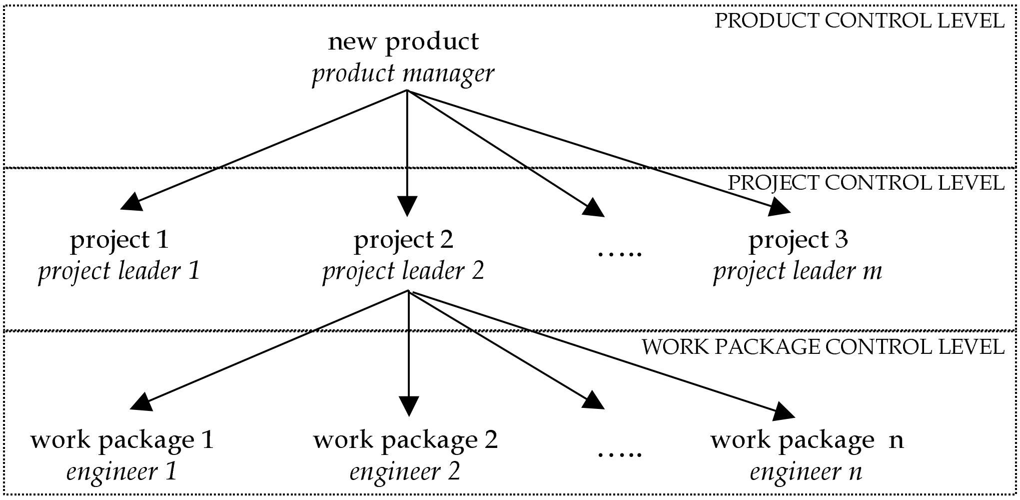


Figure 7.1 control level structure in radical NPD projects

This chapter is structured as follows. In Section 7.2 requirements for process control on the product control level are derived, followed by requirements for process control on the project and work package control level in Sections 7.3 and 7.4. The whole planning and control structure for radical NPD projects is described in Section 7.5. Finally in Section 7.6 it is explained how and why the proposed control structure resolves the planning paradox.

7.2 The product control level

In the product control level projects from the project control level are coordinated by the product manager. In production and inventory systems this is called *goods flow control*. Goods flow control (GFC) coordinates the production of goods flow controlled items that are produced in production units. GFC is also responsible for coordination with sales requirements. The sales department of a production and inventory system accepts customer orders and, based on the available capacity for producing the GFC-items the delivery date is calculated and communicated to the customer. These customer orders define precisely the number of products that are demanded and their quality or functional specifications. Sales gives the customer order to GFC and GFC realizes the customer order by giving work orders for GFC-items to production units. In radical NPD projects goods flow control is exercised at the product control level. Based on results of marketing research Marketing derives requirements for the new product and the deadline is established at which the new product must be ready. Time-to-market is very important in radical NPD projects in order to keep up with or even beat competitors. The requirements of the new product are translated into ambition levels of functional product specifications that are minimally required in order to satisfy potential customers. These minimally required ambition levels are defined here as *external ambition levels*. External ambition levels are communicated to potential customers in an early stage, thus Sales must be certain that external ambition levels can be realized by the project teams (production units). Besides external ambition levels, *internal ambition levels* are also defined. Internal ambition levels are set higher than external ambition levels of the same functional specifications. It is uncertain whether these internal ambition levels can be realized within the time available. However, if they are realized, more customers will be interested in the product and market share will increase. Because the benefits of realizing ambition levels that are higher than external ambition levels are so large, internal ambition levels are set quite high at the beginning of radical NPD projects even if the probability that these ambition levels will be realized is low. Thus, at the start of the radical NPD project the difference between internal and external ambition levels is high. During project execution more knowledge is gained on the extent to which internal ambition levels can be realized. This means that during project execution the external ambition levels can be increased if it is almost certain that project teams can realize them, but also internal ambition levels can be decreased if it is almost certain that they cannot be realized within the remaining time until the deadline. The external and internal ambition levels converge in such a way that the probability of realizing the internal ambition levels increases during project execution. In Figure 7.2 this is shown graphically.

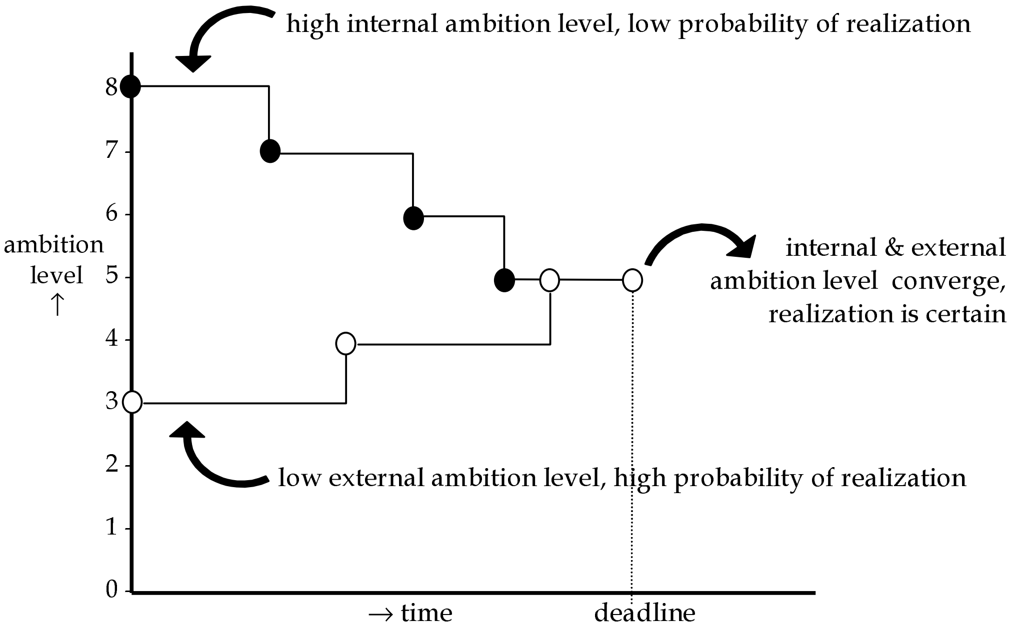


Figure 7.2 convergence of external and internal ambition levels during project execution

Thus, internal ambition levels are not static because it is uncertain at the start of the development process which (or the extent to which) ambition levels can be realized at the given deadline. Varying specifications is a characteristic that is also described in literature. For example, Bhattacharya *et al.* (1998) argue that when uncertainty surrounds a product's specifications, early product definition is not always optimal. Sometimes the information and knowledge gathered at the beginning of the radical NPD project can be obsolete at the end of the process. This may be caused by highly dynamic market situations, that change rapidly and discontinuously (Bourgeois and Eisenhardt, 1988), but it may also be caused by technological uncertainty.

The order Marketing gives to the product manager in the product control level consists of external and internal ambition levels and the product deadline. The product manager decomposes the development work into different projects that are allocated to project teams. Each project team is responsible for realizing a certain part of the functional product specifications. This means that internal ambition levels of functional *product* specifications are broken down into ambition levels of functional *project* specifications. Sometimes ambition levels of functional product specifications can be realized by a combined effort of several projects. This can be defined by describing the relationships between projects in a project structure. In this project structure the dependencies between projects are described; it thus defines which projects should work together in order to achieve the desired ambition levels. Each project is given an order to realize certain ambition levels of functional project specifications and the deadline at which the projects must be realized (top-down instruction: from product control level to project control level). After a certain time interval the projects give a progress report to the product control level (bottom-up reaction). The communication per time interval between Marketing/Sales, the product control level and the project control level can be defined by: (see also Figure 7.3):

1 the deadline at which the product must be completed;
2 external ambition levels of functional product specifications;
3 internal ambition levels of functional product specifications;
4 internal ambition levels of functional project specifications;
5 deadline at which the project must be finished;
6 progress reports;
7 estimated probability of realizing internal and external functional product specifications.

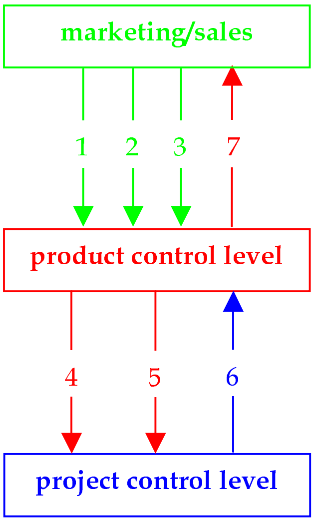
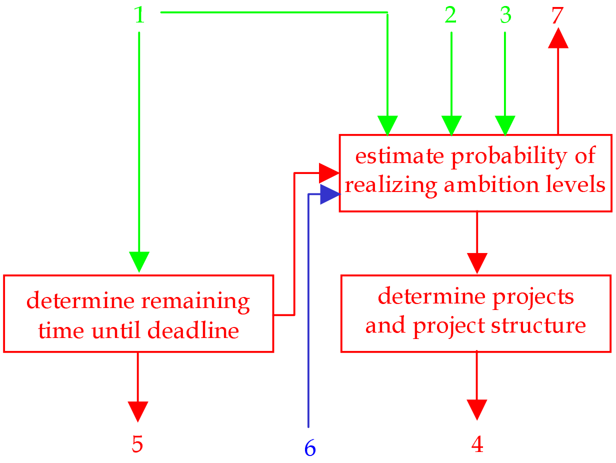


Figure 7.3 interaction between product and project control level

In the progress reports the project leaders evaluate the realized ambition levels and estimate the time required to achieve the next, higher ambition level. Based on the progress and the remaining time until the deadline the product manager can decide to adapt the internal and/or external ambition levels. And if these are adapted, so are the orders for the project control level in the next time interval. For example if the remaining time until the deadline is very short, it can be decided that the project is finished, even if the realized ambition levels are less than the internal ambition levels, but higher than or equal to the external ambition levels. Or the internal ambition level of one project is decreased, resulting in an additional burden for another project, and capacity is reallocated such that the combined results will yield a higher external ambition level. The new assignments or orders are given to the project control level for the next time interval. The decision functions described in this subsection are drawn in Figure 7.4.



Legend:
1 deadline at which the product must be realized
2 external ambition levels of functional product specifications
3 internal ambition levels of functional product specifications
4 internal ambition levels of functional project specifications
5 deadline at which the project must be completed
6 progress reports
7 estimated probability of realizing internal or external ambition levels

Figure 7.4 decision functions at the product control level of a radical NPD project

7.3 The project control level

A project team at the project control level corresponds to the concept of a production unit in production and inventory systems. But where a production unit has to deliver a certain fixed product quality and a fixed number of products within a certain time, a project team has to deliver the highest possible ambition level with a given capacity and within the deadline. The project team is self-contained with respect to how this work is done. To realize internal ambition levels of functional project specifications work packages are defined. Each work package takes part in achieving ambition levels. The sequence and dependency relationships between work packages are given in the work package structure. Each work package in the structure has a specific objective in order to realize the desired internal ambition levels (constraints). The project leader then discusses this work package structure with engineers belonging to the project team and dispatches work packages to engineers. Based on the content of work packages engineers estimate the processing time (remaining workload) of work packages. The project leader combines these estimated processing times with the available resource capacity and the deadlines given by the product control level to determine start and finish dates of each work package. By setting finish dates of critical work packages earlier than finish dates of noncritical work packages, the project leader gives priorities to work packages. It is known from cognitive psychology that a person puts forth less effort when the deadline is distal, and more effort when the deadline is proximal (Seers and Woodruff, 1997). We distinguish five interactions between project control level and work package control level (see also Figure 7.5):

8 work packages (work orders);
9 scheduling constraints;
10 (ambition level of) specification constraints;
11 progress reports;
12 processing time estimates.

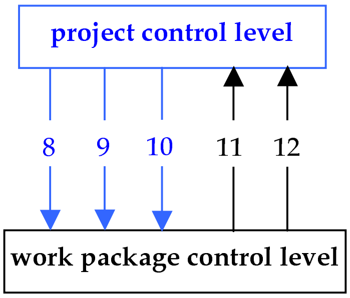


Figure 7.5: interaction between project control level and work package control level

The progress reports shown in Figure 7.5 are made by engineers after a certain time interval and are given to the project leader. It is evaluated whether ambition levels are achieved, and if not, how much time is required to do so. If this time is too long, compared to the deadlines or compared to the available capacity, it is examined whether the desired ambition level can be decreased, in order to decrease the workload. It may be possible to decrease the ambition level of one work package, at the expense of another one. Then the ambition level of the latter work package is increased, or a whole new work package originates, while the ambition level of the former work package is decreased, or the work package is even deleted from the project. When new desired ambition levels are set and the work package structure is adjusted the planned start and finish dates of the work packages can also be changed, leading to a change in priority or perceived time pressure of engineers. The cycle starts over again until the next time interval has passed and the project leader receives a new progress report. The decision functions described in this subsection are summarized in Figure 7.6. The numbered arrows reflect the three interactions between the project and product control level.

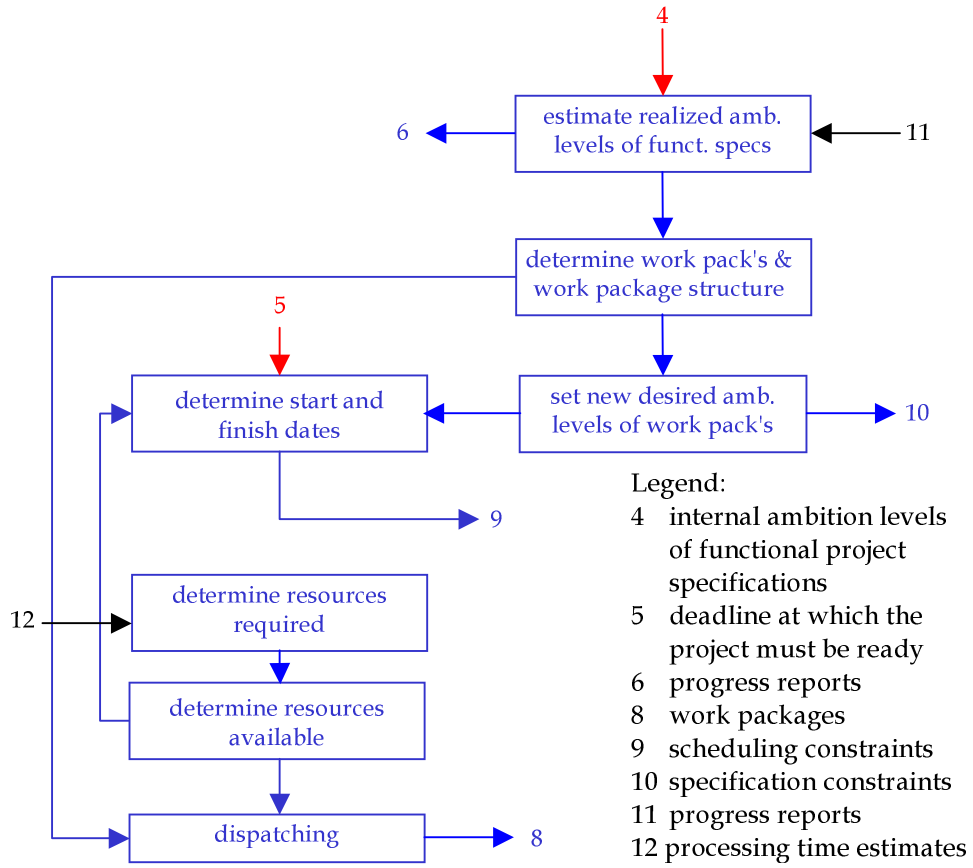


Figure 7.6 decision functions at the project control level of a radical NPD project

In Figure 7.6 two progress reports are given. One from the work package control level to the project control level and one from the project control level to the product control level. These progress reports are given after a certain time interval. The time intervals for both progress reports differ. The time interval for the communication between work package and project control level must be much shorter than the time interval for the communication between project and product control level. These time intervals can be seen as time-oriented review points or time-oriented milestones.

7.4 The work package control level

At the work package control level individual engineers are distinguished, who have to execute a certain number of work packages. These work packages are allocated by the project leader at the project control level. Work packages have operational constraints in terms of scheduling: the planned start and finish dates (and the planned flow time), and in terms of specifications: the desired ambition level of the functional project specifications. Engineers at the work package control level report to project leaders in the project control level on their progress: the realization of certain ambition levels, and on their remaining workload: the estimated processing times of work packages. The primary process for an engineer in the work package control level is shown in Figure 7.7.

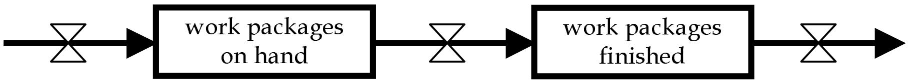


Figure 7.7: the primary process in the work package control level

Figure 7.7 shows a 'stock' of work packages that has to be executed: *work packages on hand*. An engineer chooses from this inventory a work package to work on during a certain time. This choice is influenced by the planned finish date of the work package. Work packages with a close finish date have a high priority, and therefore have a high probability to be chosen by the engineer. The extent to which this really is a 'choice' for the engineer can be influenced at the project control level, when the project leader sets the start and finish dates of work packages. The engineer works on the chosen work package for a certain time. He will change over to another work package if, for example, the planned finish date of that work package is set earlier by the project leader, or if he has trouble finding a solution for the work package he is working on. Focusing on another problem can sometimes help in finding a solution for the first problem. In cognitive psychology this is called *incubation time* (Best, 1995). Therefore it is good practice to give an engineer a few work packages to work on. After a fixed time interval, the engineer gives the project leader a progress report in which he estimates the status of his work packages. This report deals with the following questions. How many hours are spent on each work package? Are certain ambition levels of functional specifications realized? How many hours are required to realize certain ambition levels (remaining workload)? The remaining workload estimated by the engineer and the scheduling constraints on the one hand determine the perceived work pressure for the engineer in the next time interval. The project leader must control this work pressure to prevent inefficiency. The work pressure can be controlled by adjusting the planned start and/or finish dates of work packages. On the other hand the perceived work pressure is also influenced by the ambition levels of work packages. Therefore, it is of importance that the difference between external and internal ambition levels is not explicitly communicated to engineers. If engineers know the size of this difference, they may feel less pressure to try to realize these internal ambition levels. When, according to the engineers, the perceived work pressure becomes too high, they can decide to discuss this with the project leader. In turn, the project leader can decrease the work pressure by postponing the planned finish date of one or more work packages or by decreasing the ambition levels. When the work pressure is decreased, the engineer chooses once again a work package to work on during the next time interval, based on the earliest due date of his remaining work packages. The decision functions of an engineer described in this subsection are summarized in Figure 7.8. The numbered arrows reflect the five interactions between the work package and project control level.

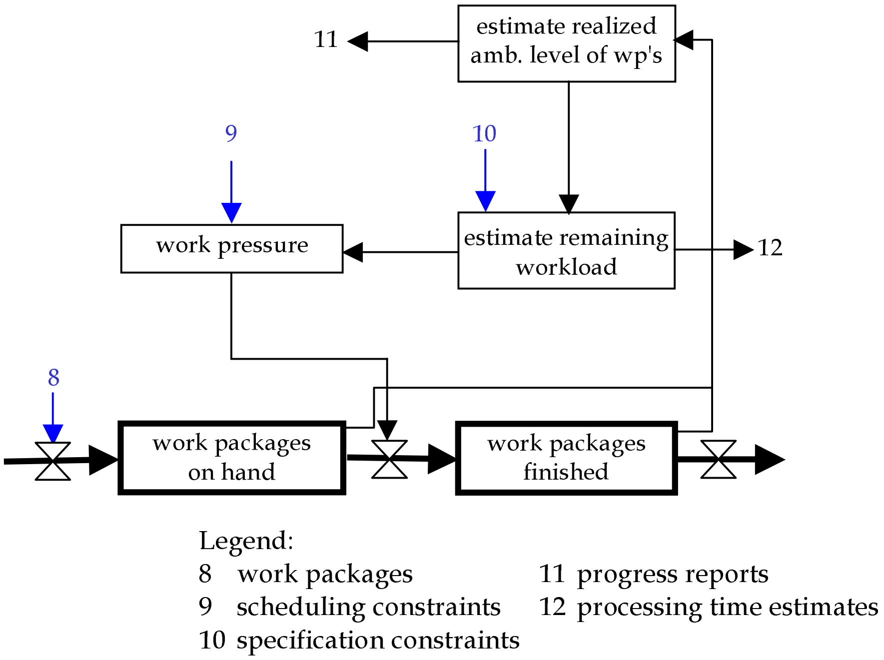


Figure 7.8 decision functions at the work package control level of a radical NPD project

7.5 The radical NPD control structure

In the previous subsections we have discussed the three control levels of radical NPD control: work package, project and product control level. The control structures of these levels are combined here to come to an overall control structure of radical NPD projects. The control structure is based on the primary process of engineers, the process of executing work packages, that is incorporated in the work package control level of the control structure. The proposed radical NPD control structure is suitable for radical NPD projects with unstable work package structures and uncertain routing and flow times. It differs from PERT and GERT, because it is not a scheduling technique. PERT and GERT can even be used in the control structure for example for scheduling the work packages at the project control level. But after each review the work package structure can be changed, thus then the PERT or GERT network must be changed too. The control structure also differs from PERT and GERT because it takes resource constraints into account. Start and finish dates of work packages are not only determined by sequencing constraints, but are influenced by the available resources. Furthermore, these start and finish dates are a mechanism for the project leader to control the sequence in which an engineer executes the work packages. Finally, the control structure differs from PERT and GERT because it assumes that ambition levels of functional specifications can be changed to realize the deadline and because both internal and external ambition levels are defined. Changing ambition levels can lead to the decrease or increase of the workload of a work package. Or it can even lead to the introduction of new work packages or the deletion of existing work packages, resulting in a change of the work package structure.

When the decision functions of the three control levels are combined in one control structure, Figure 7.9 emerges. The interaction between the three control levels that were numbered in the previous figures is given the same numbers in Figure 7.9. In this control structure the requirements are used that were suggested based on the literature and empirical study. We suggest that when the proposed control structure is used, the planning paradox, described in Chapter 3, should be resolved. In the next section this is explained.

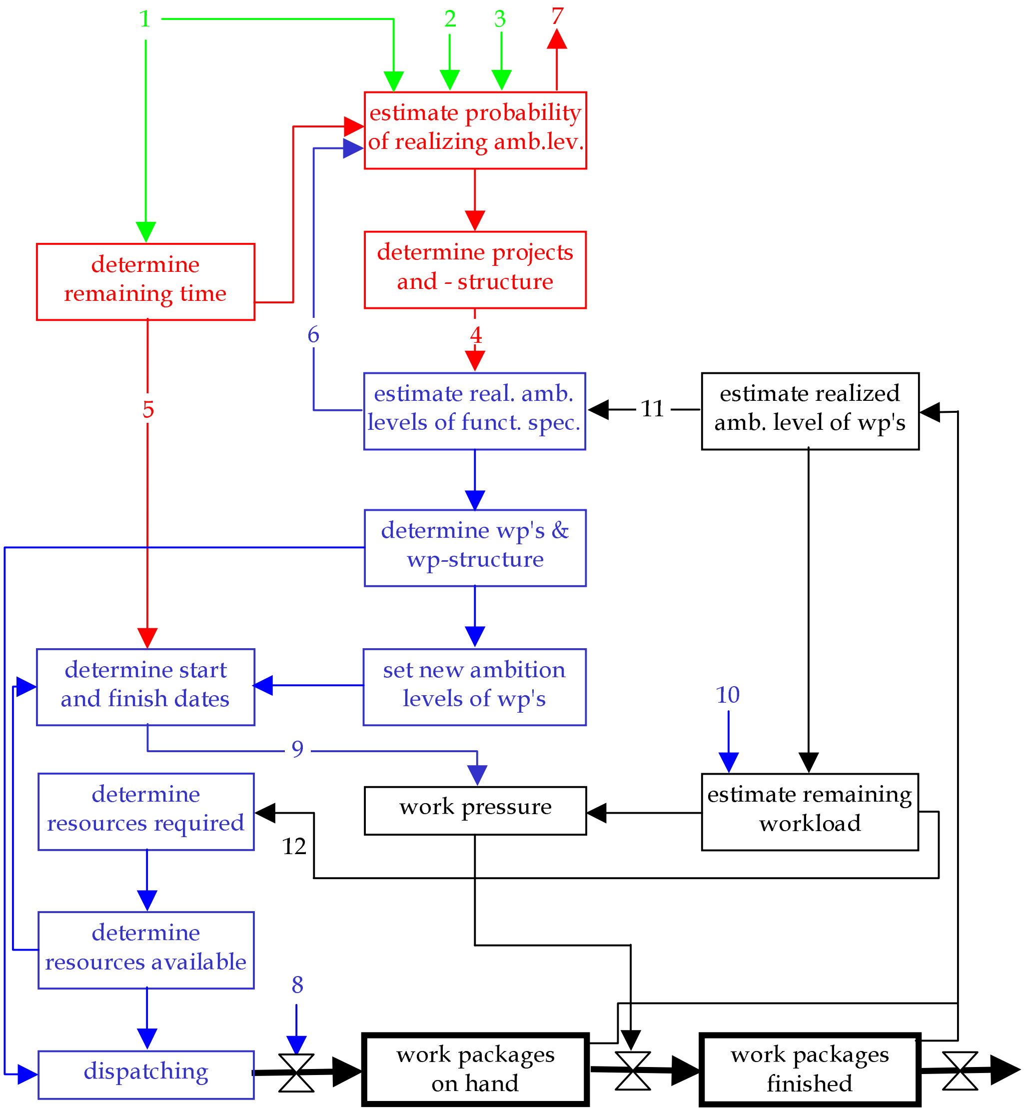


Figure 7.9 the radical NPD control structure

7.6 The planning paradox resolved

In Chapter 3 the planning paradox was introduced and discussed. The planning paradox emerges, if radical NPD projects are controlled with traditional project planning and control techniques that are not capable to deal with the uncertainty and instability of these projects. In these traditional project planning and control techniques the emerging time slack in the project plan in non-critical paths is visible to the engineers. For example in a PERT network work packages have both an earliest and a latest start date and the time between these dates is the time slack. Time slack can also be given to engineers if the time between start and finish date is longer than the time required to process a work package to absorb any unforeseen problems in the process. Furthermore, more time slack can be added to the project to compensate for the optimism of engineers estimating the processing times of work packages. However, this slack will result in engineers experiencing a low work pressure. And it is known that low work pressure leads to inefficiency (Bowers *et al.* 1997). The time slack is then consumed by inefficiency and will be lost. When a new problem (new work package) is discovered and thus suddenly more work has to be done, the planned finish date will be exceeded. For example, when in a PERT network the time slack is lost in a non-critical path, this path can suddenly become the critical path and when a new problem is discovered in this new critical path the planned finish date will be exceeded. In short, the planning paradox is: putting time slack in the project plan to prevent tardiness, but this time slack can be the cause of inefficiency and therefore cause tardiness.

In the proposed radical NPD control structure no time slack is visible to engineers. Engineers estimate processing times of work packages and the project leaders assign start and finish dates with very little or no time slack. We have shown in the empirical study in Chapter 5 that when the time pressure is very low or very high, the efficiency of engineers is lower than when the time pressure is medium-sized. Time pressure is called medium-sized if the estimated processing time is about equal to the planned flow time (difference between planned finish and planned start date), and thus when there is no time slack.

Instead of a time slack per work package, invisible and uncontrollable for the project leader after the work package is allocated to the engineer, in the radical NPD control structure no time slack exists at all. The initial internal ambition levels are so high that no time or resource slack is available. Engineers are loaded with work packages with internal finish dates such that an optimal efficiency is guaranteed. Furthermore, internal ambition levels in the beginning of the project are set sufficiently high to ensure a high work pressure. Thus, when a new problem is discovered in a project, even more resource capacity is required, so that it is obvious that the planned finish date of the work package concerned cannot be realized. Difficulties in realizing planned finish dates become clear earlier than in the traditional way of project planning and control. Engineers are 'forced' to discuss the new problem with the project leader, because as a result of this new problem, the planned finish date is not realistic. A project leader has two methods to deal with this problem. If the work package is critical for realizing an important ambition level, he can adjust the planned finish date with the time necessary for solving the new problem. If the work package is not critical, he can adjust the ambition level of the work package (or perhaps leave the new problem unsolved). Because no time slack is given to engineers, the project leader must control the difference between realized ambition levels and desired internal ambition levels and the remaining time until the deadline. During radical NPD project execution the initial ambition levels are adapted in accordance with project progress and time remaining until the deadline.

8 Conclusions and further research

8.1 Introduction

In this thesis we examined characteristics of time-constrained radical NPD projects and of the performance of engineers executing these projects. First, literature on radical NPD projects and human performance was studied, to derive these characteristics. Second, empirical studies were performed in real-life NPD projects in a large European optical equipment manufacturing firm to verify the characteristics found in literature. Third, based on the results of both literature and empirical studies, requirements for controlling radical NPD projects were derived and a control structure was proposed. In the next subsection the main findings of this research project are summarized. In Subsections 8.3 and 8.4 we evaluate this research, comment on the assumptions made in the research and describe further research on this subject.

8.2 Main research findings

In this research time-constrained radical NPD projects were examined to identify their operational characteristics. In an in-depth study of two radical software projects nineteen engineers performing work packages were observed for a period of forty-two weeks. The main findings of this in-depth study are summarized below.

Instability of work package structures

The empirical study revealed that work package structures were very unstable. The structure had to be changed on average every two or three weeks. As a consequence of these changes, it turned out that at the end of the project about 50% of the work packages in the structure were not known at the beginning of the project. The instability of work package structures seemed to be an important reason for the difficulty of the planning and control of radical NPD projects and of predicting the measure in which functional product specifications can be delivered at the project deadline.

Predictability of processing times

For all work packages, engineers made estimates of processing times (*i.e.* workload or number of man-hours required to perform the work package). Analyzing the accuracy of these processing time estimates made by engineers showed that engineers made nearly correct estimates for small work packages (work packages with an estimated workload of less than or equal to 40 hours). For larger work packages (with an estimated workload of more than 40 hours), actual processing times were significantly longer than estimated processing times. This might indicate that engineers are optimistic when they make processing time estimates. Optimism when estimating one's own future performance is a characteristic of human behavior, reported in psychology literature. Although the engineers were optimistic in the empirical study, the correlation between estimated and actual processing times was high. This means that expected actual processing times of new work packages might be determined when the estimates are made and the parameters of the relationship between estimated and actual processing times is known. In other words, processing times are not as unpredictable and uncertain as we thought at the start of the study. The effects of aha! experiences that occur when people are solving complex problems according to psychology literature did not come up in these two radical software projects. On the one hand this might imply that work packages in the software projects are not as uncertain as we expect work packages in radical NPD projects to be. But because work package structures of the software projects were very unstable and because functional product specifications were variable during project execution, we may still assume that the two software projects can be considered as radical NPD projects. On the other hand, the absence of the effects of aha! experiences might be caused by the capability of engineers in making estimates. It might be assumed that when they are experienced in processing work packages of radical NPD projects, they are able to estimate more accurately. Nevertheless, uncertainty in processing times of work packages did not seem to form a major difficulty in the planning and control of radical NPD projects in our empirical study.

Efficiency of engineers is influenced by variations in work pressure
In the empirical study we examined the relationship between work pressure and efficiency of engineers. In the two software projects the work pressure was controlled by the project leader to prevent the pressure to become too low or too high. Because of this control, the work pressure did not become very high or very low during the empirical study. Therefore it was difficult to examine the existence of a relationship between pressure and efficiency. Nevertheless, variations in work pressure did occur and a significant relationship was found between engineer efficiencies for low work pressure and medium-sized work pressure and also between engineer efficiencies for medium-sized work pressure and high work pressure. This observation supported our hypothesis that variations in work pressure influence engineer efficiency.

As a side effect of the in-depth empirical study of processing work packages by engineers, we collected information about mechanisms used to control these processes. Our observations are summarized below.

Engineers are and should be overcommitted
Engineers always had more than one work package on hand to work on. Measurements in the two software projects revealed that engineers had, on average, three work packages on hand at the same time. The reason for this approach was to avoid 'idle' time. In the empirical studies engineers were the most critical resource in product development processes. The time engineers lose, for example by waiting for information from another engineer or another process, may be valuable time in which no progress is made. When engineers have another work package on hand, this idle time may be avoided because

then they can work on this other work package. Furthermore, the number of work packages on hand influenced the work pressure perceived by engineers and consequently influenced their efficiency. Therefore, it seems sensible to allocate at least two work packages at the same time to engineers, as is also advocated by Clark and Wheelwright (1993).

Due dates of work packages are used to control work pressure

During the empirical study it was observed that planned due dates of work packages were used to control the work pressure of engineers and to assign priorities to different work packages. Of all work packages an engineer had on hand, the one with the closest due date had the highest priority. When work pressure caused by the work packages and their due dates became too high, due dates of one or more work packages were postponed, or in some cases some work packages were deleted from the project plan. This observation was supported by the measurements in the empirical study of planned and actual flow times. A flow time is the number of days between the day an engineer starts with a work package and the day he finishes this work package. It was measured that actual flow times were barely related to planned flow times. Actual flow times were much longer than planned. The reasons for this were explained above. Furthermore during the empirical study we observed that planned flow times were calculated assuming that engineers had only one work package on hand. Since we measured that engineers had on average three work packages on hand, it was not surprising that actual flow times were much longer than planned flow times.

Internal ambition levels versus external ambition levels

In the firm in which the empirical study was performed we observed that to develop a new product, different project teams were formed. Each project team had the objective to realize a part of the functional product specifications. Objectives were defined as *internal ambition levels* of functional specifications. An ambition level indicated the measure in which a functional specification was realized. The higher the ambition level, the 'better' the functional specification and the more satisfied potential customers would be. Because of the uncertainty in radical NPD projects, it was not certain which ambition levels could be realized at the project deadline. Therefore, at the start of development projects, potential customers were given *external ambition levels* of functional specifications. These external ambition levels were lower than internal ambition levels, but the probability of realizing the external ambition levels was much higher than that of internal ambition levels. Because the risk of not realizing external ambition levels was low, these levels could be communicated to potential customers. Internal ambition levels were set higher to increase work pressure. However, these internal ambition levels increased work pressure only if external ambition levels were not made explicit. If engineers were aware of the difference between external and internal ambition levels, they might perceive a lower work pressure. During project execution it might become certain that internal ambition levels could not be realized within the time given. Then these levels could be set lower (until external ambition levels were reached). This approach caused every project team to try to realize the highest possible ambition level (and consequently the highest customer satisfaction). Instead of a time slack that could be given to engineers for unforeseen future problems, project leaders were given another type of slack, resulting from the difference between internal and external ambition levels.

Based on the results of the in-depth empirical study and the observations of the way in which development processes were controlled, requirements for NPD project control were derived. Furthermore a NPD control structure was proposed. The proposed control structure consists of three levels: the work package control level, the project control level and the product control level. In the work package control level work packages are performed by engineers. Engineers, having more work packages on hand at the same time, select the work package with the closest planned finish date to work on. After a fixed time period engineers review the progress made, estimate the realized ambition level of work packages, and for the next time period, they estimate the remaining workload and the perceived work pressure. This information collected by engineers is communicated to the project leader in the project control level. In the project control level the project leader controls the progress of the project in order to realize the project due date. To do so, the project leader makes sure that the work pressure of engineers is sufficiently high (by allocating work packages) to support the efficiency of engineers. When the work pressure becomes too high, the project leader may decide to adjust ambition levels of work packages or to postpone planned finish dates of work packages. These measures may not be sufficient to decrease work pressure. Then the project due date is endangered and the product manager in the product control level will be notified. In the product control level the product manager may decide to adjust the ambition levels of the project. As a consequence either the ambition levels of another project are increased to compensate for the decrease of ambition levels of the first project, or, if this is not possible, the internal ambition levels of the product are decreased. This means that the 'slack' that exists between external (required) and internal (desired) ambition levels of the product is decreased. But, the project due date is preserved.

The proposed control structure may still use some planning and control techniques for the detailed planning and scheduling of work packages. The control structure does not exclude the use of traditional techniques like PERT or GERT. These techniques may be used on a short term by project leaders to determine which work packages have to be done first. However, because the work package structure of radical NPD projects changes so often, it does not seem efficient to make PERT or GERT networks for the whole project (long term).

8.3 Evaluation of the research

8.3.1 Strengths and weaknesses

In this section the research is evaluated. In this subsection we will describe strengths and weaknesses of the research. In the next subsection we will discuss the impact of assumptions made in this research.

Strengths of this research

- In the empirical study the individual work package is the unit of research. This is not common in project management research because of the great difficulty in collecting empirical data at this level of observation. Past empirical research on development projects usually considered whole projects or development strategies. Furthermore studying the dynamics of the work package structure has not been done before.
- Assumptions made in traditional project planning and control techniques were challenged by presenting real-life data.
- Empirical data were collected in a time-paced context, in which little prior research has been done. Furthermore, the dimension of the in-depth empirical study was large: empirical data were collected in a longitudinal study (forty-two weeks), and detailed data were collected from nineteen engineers from two different projects.

Weaknesses of this research

- Empirical data were collected from two radical NPD projects, that were both characterized as software development projects. No data were collected from projects in which other technological disciplines were required.
- Data were collected in a controlled situation. The control structure that we proposed is for a large part inspired by observations made during the empirical study and thus based on the experience of project leaders and engineers. Therefore the effect of using the control structure could not be examined empirically in the study. Furthermore, no field test has been performed to verify whether the proposed control structure is suitable for the planning and control of radical NPD projects. Though the control structure seems suitable for the firm involved in the empirical studies, the control structure has not been tested in another firm.
- To examine psychological constructs like optimism, pressure and efficiency of engineers we defined indirect measures based on project data. We did not compare the measures with the perceptions of engineers. For example, if a high work pressure was measured, the engineer in question was not asked to describe how he perceived his work pressure. It might be that measurement and perception not always corresponded. Engineers were not actively involved in the empirical study, to avoid interference by the study (like reporting more working hours, because they know they are observed).

8.3.2 Impact of assumptions

A number of assumptions were made in this research project. These assumptions seemed valid in the radical NPD projects in which we performed the empirical research. Therefore some comments are required on the generalization of our conjectures and the proposed control structure for radical NPD projects. The following assumptions are considered:

- human resources are scarce, financial resources are not scarce;
- application and basic software NPD projects are the least uncertain compared to NPD projects in other technical disciplines;
- engineers estimate honestly.

First, the empirical research showed that the financial resources of NPD were not the bottleneck in performing theses projects. However, the human resources (engineers) were very scarce. Radical NPD projects require technically high-educated people and currently, many vacancies for these people exist on the Dutch job market and in particular in the company of our research. When more engineers work on a project, more work packages can be performed and higher ambition levels can be realized. But when the project team consists of more engineers, also more time has to be spent on reviewing the work, meetings, etc. So when human resources are not scarce, probably a trade-off exists between adding more engineers to the project team and losing development time for communication between engineers. The former results in an increase of the available man-hours that can be spent on work packages and the latter results in a decrease of the available man-hours. In this situation the control structure may be expanded by including the decision to allow more engineers in the project team.

Second, the empirical research was performed in two software NPD projects. The engineers in the two software teams were already accustomed to estimate processing times every week, which was one of the reasons why these teams were included in the research. In the firm these two software projects are known to be the least uncertain, compared to other radical NPD projects. Thus, the characteristics of radical NPD projects and of the performance of engineers executing these projects, which we found in the literature, were verified in the empirical research in these least uncertain radical NPD projects. We assume that these characteristics are encountered also in other technical disciplines in the firm, where the NPD projects are more uncertain. The control structure does not have to be changed when the projects are more uncertain. When there is more uncertainty, it is more difficult for engineers to estimate processing times accurately, and it is more difficult to predict the ambition levels of functional specifications that can be realized at the deadline. It might be necessary to increase the number of time-oriented review points when uncertainty increases, to increase communication and knowledge transfer between engineers, project leaders and the product manager. But by increasing the number of time-oriented review points, the control structure that we proposed is still valid.

Third, the culture of the firm in which the radical NPD projects are performed is of importance. Engineers are required to estimate processing times of work packages. These estimates are the basis for the project plan in which planned start and finish dates of work packages are defined. In turn, these planned start and finish dates determine the work pressure of engineers. If engineers are not encouraged to estimate honestly, for example, if they are penalized when estimates are not correct or when actual processing times are longer than estimated, they might begin to estimate 'safely'. In other words, they might include a time slack in their estimates in order to gain more time from the project leader. This will decrease the work pressure and thus the efficiency of engineers. This behavior can even result in the exceeding of the planned finish date that might encourage engineers to put even more time slack in their estimates the next time. Thus it seems important that engineers are honest, and are encouraged to be honest, when they estimate processing times and that they are not penalized if the actual processing times are longer than estimated.

8.4 Further research

Herewith I come to the end of this research. I hope that the studies and the control structure presented in this thesis will stimulate research by others. The topic is of great importance, because not much research has been done on work package structures in radical NPD projects and on the combination of mechanistic and organic approaches in controlling radical NPD projects. Furthermore, from a managerial point of view the control of these projects is of importance, to maximize product quality (or external ambition levels) within the given due date, and consequently, to maximize customer satisfaction. Further research could be centered around three themes: the instability of work package structures in radical NPD projects, the uncertainty of processing times, and the implementation of the proposed control structure.

Instability of the work package structure

The in-depth empirical study revealed that work package structures of the two software radical NPD projects were highly unstable. A large part of the uncertainty in radical NPD projects seemed to be caused by the emergence of new work packages that were not foreseen and planned in the beginning of the project. Because of the importance of the instability of work package structures for the control of radical NPD projects, more (empirical) research in this area is required. The following subjects are of interest:

- Instability of work package structures in other real-life projects.

In the previous section we discussed that the two software projects included in the empirical study were considered to be the least uncertain compared to other projects in the firm. It is interesting to examine the instability in other disciplines, like mechanical, electronical, optical engineering, as well.

- Origin of new work packages that emerge during project execution.

Besides knowing that work package structures are unstable, it is also necessary to examine the causes for this instability. To plan and control radical NPD projects it can be helpful to understand and know why new work packages are discovered. New work packages might be caused by changing functional specifications, but they might be caused also by errors (rework) discovered in work packages that were performed earlier in the project or even in former projects (see for example Sander and Brombacher, 2000). Examining the origin of new work packages might make their emergence more predictable in statistical sense. When a reliable prediction can be made of the number of work packages that will emerge during project execution, the workload of the project can be estimated more accurately.

- Interarrival times of changes in the work package structure.

In this research the interarrival times of structure changes of only two software projects were measured. It is interesting to measure this for other, more uncertain projects as well. The interarrival time can be used to determine the horizon over which detailed project plans can be made and to determine the frequency of time-oriented reviews. It is likely that the more uncertain the project, the shorter the interarrival times of work package structure changes is, but this assumption needs to be examined.

Uncertainty of processing times

It was shown in the empirical study that especially for large work packages (with an estimated workload of more than 40 man-hours), engineers seemed to be optimistic (actual processing times were significantly longer than estimated). However, the difference between actual and estimated processing times was small and did not suggest that processing times are uncertain and difficult to estimate. This was a surprising result, contrary to what we expected based on the literature study described in Chapter 2. Therefore it is interesting to examine whether engineers are able to make accurate processing time estimates of work packages in other radical NPD projects as well. Furthermore it is interesting to examine why they are able or unable to make accurate estimates. For example, is the accuracy of estimates dependent on the experience of engineers, or on the engineering discipline (software, mechanical, electronical engineering, etcetera)? Moreover, studying the statistical relationship between estimated and actual processing times is of importance to determine the probability that a project will be finished within a certain time.

Implementation of the proposed control structure

Some elements of the proposed control structure were already common practice in the firm in which the empirical studies were performed. It is interesting to find other firms in which radical NPD projects are performed and to analyze how these projects are planned and controlled. Then differences with the proposed control structure can be identified and hypotheses can be defined about the performance of these firms. For example, if firms use traditional project planning and control techniques, we expect that projects will not be finished in time. Furthermore, it may be possible to introduce the proposed control structure to radical NPD project teams that have been using only traditional project planning and control techniques. When the proposed control structure is used for a radical NPD project for the first time, engineers and project leaders may evaluate the control structure compared to their former planning and control approach. This evaluation will provide important information to verify the extent to which the proposed control structure is suitable for radical NPD projects.

References

Adler, P.S., A. Mandelbaum, V. Nguyen and E. Schwerer, 1995, From project to process management: an empirically-based framework for analyzing product development time, *Management Science*, 41, 458-484.

Amundson, S.D., 1998, Relationships between theory-driven empirical research in operations management and other disciplines, *Journal of Operations Management*, 16, 341-359

Andreasen, M.M. and L. Hein, 1987, *Integrated product development*, Springer-Verlag, Berlin.

Bazerman, M.H., A.E. Tenbrunsel and K. Wade-Benzoni, 1998, Negotiating with yourself and losing: making decisions with competing internal preferences, *Academy of Management Review*, 23, 225-241.

Beer, S., 1959, What has cybernetics to do with OR?, *Operational Research Quarterly*, 10, 1-21.

Berden, T.P.J., A.C. Brombacher and P.C. Sander, 2000, The building bricks of product quality: an overview of some basic concepts and principles, *International Journal of Production Economics*, 67, 3-15.

Bertrand, J.W.M., J.C. Wortmann and J. Wijngaard, 1990, *Production control, a structural and design oriented approach*, Elsevier, Amsterdam.

Bertrand, J.W.M. and D.R. Muntslag, 1993, Production control in engineer-to-order firms, *International Journal of Production Economics*, 30-31, 3-22.

Best, J.B., 1995, *Cognitive Psychology*, West Publishing Company, St. Paul, Minneapolis.

Bhattacharya, S., V. Krishnan and V. Mahajan, 1998, Managing new product definition in highly dynamic environments, *Management Science*, 44, 50-64.

Boctor, F.F., 1990, Some efficient multi-heuristic procedures for resource-constrained project scheduling, *European Journal of Operations Research*, 49, 3-13.

Boer, R. de, 1998, *Resource-constrained multi-project management, a hierarchical decision support system*, Thesis University of Twente.

Bourgeois, L.J. and K. Eisenhardt, 1998, Strategic decision processes in high velocity environments: four cases in the microcomputer industry, *Management Science*, 34, 816-835.

Bowers, C.A., C.C. Braun and B.B. Morgan, Jr., 1997, Team workload: its meaning and measurement, M.T. Brannick, E. Salas and C. Prince (eds), *Team Performance Assessment and Measurement: theory, research, and applications*, Lawrence Erlbaum Associates, Inc., Publishers, New Jersey, 85-108.

Brown, S.L. and K.M. Eisenhardt, 1995, Product development: past research, present findings, and future directions, *Academy of Management Review*, 20, 343-378.

Buehler, R., D. Griffin and M. Ross, 1994, Exploring the "planning fallacy": why people underestimate their task completion times, *Journal of Personality and Social Psychology*, 67, 366-381.

Clark, K.B. and T. Fujimoto, 1991, *Product development performance, strategy, organization, and management in the world auto industry*, Harvard Business School Press, Boston.

Clark, K.B. and S.C. Wheelwright, 1993, *Managing new product and process development: text and cases*, The Free Press, New York.

Cooper, R.G., 1994, Third-generation new product processes, *Journal of Product Innovation Management*, 11, 3-14.

Cyert, R.M. and J.G. March, 1963, *A behavioral theory of the firm*, Prentice-Hall, Englewood Cliffs.

Davis, E.W. and J.H. Patterson, 1975, A comparison of heuristic and optimum solutions in resource-constrained project scheduling, *Management Science*, 21, 944-955.

Dawson, R.J. and C.W. Dawson, 1998, Practical proposals for managing uncertainty and risk in project planning, *International Journal of Project Management*, 16, 299-310.

Demeulemeester, E., 1995, Minimizing resource availability costs in time-limited project networks, *Management Science*, 41, 1590-1598.

Dewar, R.D. and J.E. Dutton, 1986, The adoption of radical and incremental innovations: an empirical analysis, *Management Science*, 32, 1422-1433.

Eisenhardt, K.M. and B.N. Tabrizi, 1995, Accelerating adaptive processes: product innovation in the global computer industry, *Administrative Science Quarterly*, 40, 84-110.

Ettlie, J.E., W.P Bridges and R.D. O'Keefe, 1984, Organization strategy and structural differences for radical versus incremental innovation, *Management Science*, 30, 682-695.

Fox, W.M., 1988, *Effective Group Problem Solving*, Jossey-Bass Publishers, San Francisco.

Galbraith, J., 1973, *Designing complex organizations*, Addison-Wesley Publishing Company, Inc., Reading, Massachusetts.

Gersick C.J.G., 1988, Time and transition in work teams: toward a new model of group development, *Academy of Management Journal*, 3, 9-41.

Gersick C.J.G., 1994, Pacing strategic change: the case of a new venture, *Academy of Management Journal*, 9, 9-45.

Gummesson, E., 1991, *Qualitative Methods in Management Research,* Sage Publications, Newbury Park, California.

Handfield, R.B., and S.A. Melnyk, 1998, The scientific theory-building process: a primer using the case of TQM, *Journal of Operations Management*, 16, 321-339.

Hanssen, R.W., 2000, *Concurrent engineering vanuit beheersingsperspectief,* Thesis Technische Universiteit Eindhoven (in Dutch).

Huchzermeier, A. and C.H. Loch, 2001, Project management under risk: using the real options approach to evaluate flexibility in R&D, *Management Science,* 47, 85-101.

Icmeli Tukel, O. and W.O. Rom, 1998, Analysis of the characteristics of projects in diverse industries, *Journal of Operations Management*, 16, 43-61.

Kahneman, D. and A. Tversky, 1979, Intuitive prediction: biases and corrective procedures, *TIMS Studies in Management Science*, 12, 313-327.

Karau, S.J. and J. R. Kelly, 1992, The effects of time scarcity and time abundance on group performance quality and interaction process, *Journal of Experimental Social Psychology*, 28, 542-571.

Keizers, J.M., 2000, *Subcontracting as a capacity management tool in multi-project repair shops*, Thesis Technische Universiteit Eindhoven.

Keys, P., 1991, *Operational research and systems: the systemic nature of operational research,* Plenum Press, New York.

Kidd, J.B., 1991, Do today's projects need powerful network planning tools?, *International Journal of Production Research*, 29, 1969-1978.

Kim, S.Y. and R.C. Leachman, 1993, Multi-project scheduling with explicit lateness costs, *IIE Transactions*, 25, 34-44.

Kolisch, R., A. Sprecher and A. Drexl, 1995, Characterization and generation of a general class of resource-constrained project scheduling, *Management Science*, 41, 1693-1703.

Krishnan, V., S.D. Eppinger and D.E. Whitney, 1997, A model-based framework to overlap product development activities, *Management Science*, 43, 437-451.

Kurtulus, I. and E.W. Davis, 1982, Multi-project scheduling: categorization of heuristic rules performance, *Management Science*, 28, 161-172.

Lai, L.S.L., 1997, A synergistic approach to project management in information systems development, *International Journal of Project Management*, 15, 173-179.

Latham, G.P. and G.H. Seijts, 1999, The effects of proximal and distal goals on performance on a moderately complex task, *Journal of Organizational Behavior*, 20, 421-429.

Lawrence, S.R. and T.E. Morton, 1993, Resource-constrained multi-project scheduling with tardy costs: comparing myopic, bottleneck, and resource pricing heuristics, *European Journal of Operational Research*, 64, 168-187.

Leachman, R.C. and J. Boysen, 1985, An aggregate model for multi-project resource allocation, B.V. Dean (ed.) *Project management: methods and studies*, Elsevier Science Publishers, Amsterdam, 43-64.

Li, K.Y. and R.J. Willis, 1992, An iterative scheduling technique for resource-constrained project scheduling, *European Journal of Operational Research*, 56, 370-379.

Lipshitz, R. and O. Strauss, 1997, Coping with uncertainty: a naturalistic decision-making analysis, *Organizational Behavior and Human Decision Processes*, 69, 149-163.

Loehle, C., 1994, A critical path analysis of scientific productivity, *Journal of Creative Behavior*, 28, 33-47.

McDermott, C., 1999, Managing radical product development in large manufacturing firms: a longitudinal study, *Journal of Operations Management*, 17, 631-644.

Meal, H.C., 1984, Putting production decisions where they belong, *Harvard Business Review*, 62, 102-111.

Mendenhall, W. and T. Sincich, 1988, *Statistics for the engineering and computer sciences*, Dellen Publishing Company, San Francisco.

Meredith, J.R. and S.D. Mantel, jr., 1989, *Project management, a managerial approach*, John Wiley & Sons, Singapore.

L. Mertens, 26 Feb 1999, Product Management Plan Software Layout S700

Moore, C.M., 1987, Group techniques for idea building, *Applied Social Research Methods Series*, 9, Sage Publications, Beverly Hills.

Muntslag, 1993, *Managing customer order driven engineering, an interdisciplinary and design oriented approach*, Thesis Technische Universiteit Eindhoven.

Muthén, B.O. and A. Satorra, 1995, Complex sample data in structural equation modeling, *Sociological Methodology*, 25, 267-316.

Neumann, K. and U. Steinhardt, 1979, *GERT networks and time-oriented evaluation of projects*, Springer-Verlag, Berlin.

Özdamar, L. and G. Ulusoy, 1995, A survey on the resource-constrained project scheduling problem, *IIE Transactions*, 27, 574-586.

Parkinson, C.N., 1957, *Parkinson's law and other studies in administration*, Mifflin, Boston.

Petri, H.L., 1981, *Motivation: theory and research*, Wadsworth, Inc., Belmont.

Prasad, B., 1996, Toward definitions of a concurrent product design, development, and delivery (PD^3) system, *Concurrent Engineering*, 4, 102-109.

Project Management Institute (PMI), 1996, *A guide to the project management body of knowledge*, PMI, Upper Darby USA.

Reed, S.K., 1988, *Cognition, theory and applications*, Brooks/Cole Publishing Company, Pacific Grove, California.

Repenning, N.P., 2000, A dynamic model of resource allocation in multi-project research and development systems, *System Dynamics Review*, 16, 173-212.

Reiss, G., 1996, *Programme management demystified*, Chapman & Hall, London.

Sander, P.C. and A.C. Brombacher, 2000, Analysis of quality information flows in the product creation process of high volume consumer products, *International Journal of Production Economics*, 67, 37-52.

Schneeweiss, C., 1999, *Hierarchies in Distributed decision making*, Springer-Verlag, Berlin.

Seers, A. and S. Woodruff, 1997, Temporal pacing in task forces: group development or deadline pressure, *Journal of Management*, 23, 169-187.

Sheremata, W.A., 2000, Centrifugal and centripetal forces in radical new product development under time pressure, *Academy of Management Review*, 25, 389-408.

Shtub, A., J.F. Bard and S. Globerson, 1994, *Project management: engineering, technology, and implementation*, Prentice Hall International, Englewood Cliffs.

Smith, P.G. and D.G. Reinertsen, 1995, *Developing products in half the time*, Van Nostrand Reinhold, New York.

Stalk, J. jr. and T.M. Hout, 1990, *Competing against time: how time-based competition is reshaping global markets*, The Free Press, New York.

Strauss, A. and J. Corbin, 1990, *Basics of Qualitative Research: grounded theory procedures and techniques*, Sage Publications, Newbury Park, California.

Tatikonda, M.V. and S.R. Rosenthal, 2000, Successful execution of product development projects: balancing firmness and flexibility in the innovation process, *Journal of Operations Management*, 18, 401-425.

Taylor, B.W. and L.J. Moore, 1980, R&D project planning with Q-GERT network modeling and simulation, *Management Science*, 26, 44-59.

Thompson, J.D., 1967, *Organizations in Action*, McGraw-Hill Inc., New York.

Tripoli, A.M., 1998, Planning and allocating: strategies for managing priorities in complex jobs, *European Journal of Work and Organizational Psychology*, 7, 455-476.

Turner, J.R. and R.A. Cochrane, 1993, Goals-and-methods matrix: coping with projects with ill defined goals and/or methods of achieving them, *International Journal of Project Management*, 11, 93-102.

Ulrich, K.T. and S.D. Eppinger, 1995, *Product design and development*, McGraw-Hill Inc., New York.

Van de Wakker, A.M., 1993, *Throughput time control and due date reliability in tool & die shops*, Thesis Technische Universiteit Eindhoven.

Webb, A., 1994, *Managing Innovative Projects*, Chapman & Hall, London.

Wheelwright, S.C. and K.B. Clark, 1995, *Leading product development: the senior manager's guide to creating and shaping the enterprise*, The Free Press, New York.

Wickens, C.D., 1984, Processing resources in attention, R. Parasuraman and D.R. Davies (eds), *Varieties of Attention*, Academic Press Inc., Orlando, Florida.

Appendix

Clusters of causes resulting from using the NGT with engineers and project leaders working on radical NPD projects

The remarks the engineers and project leaders made, are not adjusted and copied directly from the sheets used during the study. Because the causes were generated in small groups of three people, some causes can be written down more than once. Between brackets behind the cluster name the amount of times the cluster was chosen as 'most important' is given.

A vague specifications (3)
- unclear specifications
- quality of specifications not sufficient
- unclear specifications
- unclear specifications: budgets are finished up by two projects
- unclear specifications
- vague specifications

B varying specifications (1)
- no constant specifications
- changing specifications
- specifications of the project change or those of other projects
- specifications are definite too late
- specifications change because of: learning from a proto machine, exchange between overlay and throughput
- specifications change/exchange between projects
- specifications are not present at the start of the project
- new (derived) specifications arrive during development and implementation
- no specification control

C specification procedure (1)

- underlying specifications are not achieved or cannot be tested
- old design problems bite back
- the firm gives unclear specifications to third parties
- I accept unclear assignments!

D inaccurately estimating the amount of work (4)

- optimistic planning of the project leader and the project member
- planning too optimistically
- go along with an unrealistic planning

E unplanned activities (1)

- many small disturbances (questions, telephone calls, coaching, etc.)
- does not say "no" often enough (letting yourself get disturbed)
- focusing on immediate questions that were unforeseen
- many interruptions during work (discussions, telephone calls, questions: "if you are disturbed every ten minutes, you get exactly nothing done")
- moving to another building interrupts development
- unforeseen problems can cost a lot of time
- unreliable delivery times
- escalation outside the project

F balancing between projects (2)

- balancing between projects is wrong → one project is faster than another
- balancing between different disciplines is wrong (for example electronics vs. mechanics, tooling vs. design)
- communication between projects is not working: volume control and interface agreements
- planning of Electronics/Software/Architecture is not known
- planning of System Engineering is not known

G not working according to the project-approach: planning and tracking (0)

- too much operational work → not much time for project planning
- personal planning is not up to date
- no tracking → in the next planning the same mistakes are made
- unclear project planning
- lacking of an integral project planning
- admitting too late that the planning will be exceeded
- planning of complementary projects is not known

H non-human resources (1)
- no test tooling
- availability of machines is too limited
- too little machine time

I quantity of human resources (0)
- several projects with one person, pushing priorities aside
- unclear net availability
- claim and allocation procedure does not work: must be more stable
- resources disappear or do not come/are taken back
- resources are shredded across projects
- key persons leave projects too soon
- aftermath of former projects
- sickness of key persons
- allocate resource claim
- no or too little resources
- uncertain resource allocation

J quality of human resources (0)
- education of people in the project
- training of elementary skills in own discipline
- quality of project members (junior members without coaching)
- inexperienced project members: technology and planning
- too little transfer of knowledge
- resource to which the job is allocated, appears not to be the right resource
- too little resources or too inexperienced

Summary

In this thesis we focus on time-constrained radical new product development (NPD) projects. We investigate characteristics of radical NPD projects and of engineers that perform development processes in these projects. Furthermore, we study the control of radical NPD projects in relation to these characteristics. We also suggest a control structure for radical NPD projects, consisting of the main decision functions, their goals, their (inter)relationships, and the organizational positions to which the decision functions should be assigned.

Chapter 2 reports on the results of a literature research in NPD projects, in particular characteristics of time-constrained radical NPD projects and of techniques for planning and controlling these projects. From an operational control perspective, technological uncertainty has been identified as an important characteristic of radical NPD projects. It influences the execution of development processes in several ways:

- functional product specifications are not fixed at the beginning of the project, but these specifications may be changed during project execution, leading to the addition or deletion of work packages and thus leading to a variable work package structure;
- unplanned or unforeseen work packages may be added to planned work packages when during the processing of a work package a new problem emerges that is so extensive that a new work package needs to be defined;
- processing times of work packages are difficult to predict because solutions of design problems contained in work packages is often found suddenly.

Previous research in managing uncertain situations reveals that both mechanistic and organic approaches are required to deal with uncertainty. Existing project planning and control techniques, like PERT and GERT, have a mechanistic nature. These traditional techniques assume stable work package structures, known (stochastic distributions of) processing times of work packages, and fixed deliverables. Furthermore, traditional techniques largely ignore human aspects of performing development processes.

In Chapter 3 we hypothesize what might occur when radical NPD projects are planned and controlled with traditional, mechanistic techniques. From psychology literature, we have identified two characteristics of particular importance. First, people tend to be optimistic when they estimate processing times of work packages they have to perform. Second, human performance seems to be dependent on perceived work pressure. Performance degrades at low and high levels of work pressure and a medium-sized level of work pressure results in the highest performance. In this chapter we suggest how these characteristics influence development processes. To compensate for uncertainty and optimism in NPD projects, traditional techniques lead to project plans with time and resource slack to account for events unknown at the start of the project. However, if time slack is put in a NPD project plan, engineers will perceive low work pressure and consequently their performance will deteriorate. The time slack would be lost, because it is consumed by inefficiency. If a new problem occurs and more (unforeseen) work has to be done, little or no slack would be left and the planned finish date would be exceeded. We therefore hypothesize that application of traditional project planning and control techniques to NPD projects will lead to a planning paradox; time slack put in the project plan to prevent it from running late, will induce human behavior that results in the project to run late. Using these techniques to plan and control radical NPD processes is questioned.

In Chapter 4 a case description is given of a large European optical equipment manufacturing firm. In this firm we performed empirical studies in which we examined radical NPD projects in real life. But first we had to verify whether the development projects in this firm could be characterized as radical NPD projects. To do so we performed three small-scale empirical studies that are described in Chapter 5. These studies used data that were already available in the firm. If the projects could be characterized as radical NPD projects an in-depth empirical study in the same firm would be justified. The first small-scale study showed that planned flow times are consistently much shorter than actual flow times, which can be considered as an indication of optimism and/or the emergence of unplanned work packages. In the second small-scale study, a group of engineers and project leaders was asked to identify reasons why radical NPD projects are difficult to plan. The two most important reasons mentioned were: changing functional specifications (leading to an unstable work package structure) and inaccurate or optimistic estimates of capacity required. In the third small-scale study, the number of work packages of a mechanical NPD project was measured every two weeks. The study showed that this number of work packages changed very frequently, thereby providing further evidence that work package structures in radical NPD projects tend to be unstable. The results of these studies were in accordance with the characteristics of radical NPD projects and human behavior found in the literature. This justified a more in-depth empirical study in this firm.

In Chapter 6 we report on this in-depth study concerning the following conjectures regarding characteristics of radical NPD projects and human behavior:

- unplanned work packages emerge during the project;
- engineers are optimistic when estimating processing times;
- a curvi-linear relationship exists between perceived work pressure of engineers and their efficiency.

The research was performed on two radical NPD software projects. Software NPD projects are considered to be the least uncertain compared to NPD projects in other technical disciplines of the firm. Therefore we may assume that if support for the conjectures is found in these projects, they will pertain also in other NPD projects. For a period of 42 weeks 19 engineers of these two projects were asked to estimate the number of hours they worked on each work package and the number of hours that was required to finish the work package. These estimates and the actual hours and flow times were used to test the conjectures. For all conjectures we found support in these real-life projects. First, work package structures of the projects changed on average every two or three weeks. Second, a linear relationship seemed to exist between estimated and actual processing times. Although processing times were not as uncertain as expected, actual processing times were consistently longer than estimated processing times, indicating that engineers might be optimistic. Finally, significantly higher productivity was found for medium-sized work pressure compared to low and high work pressure. As a byproduct of the empirical research, also information was gained about the way NPD projects were being controlled in the development department. This has been input to the design of a control structure for radical NPD projects.

In Chapter 7, based on results of the empirical studies, requirements for controlling radical NPD projects are derived and a control structure for these projects is proposed. The control structure should help solving the radical NPD control problem: given certain minimal objectives (external ambition levels) regarding functional product specifications, given the deadline at which the new product has to be available, and given the resource capacity, how should we: 1) determine desired (internal) ambition levels of functional specifications of the product; 2) define work packages for engineers; 3) vary ambition levels of work packages and/or work package structures; 4) allocate work packages to engineers. The proposed control structure resolves the planning paradox. Traditional project planning and control techniques, like PERT and GERT, assume fixed deliverables and advocate putting time slack in the project plan that may be used to solve unplanned, new problems that arise. This time slack however leads to a low work pressure and thus a low efficiency. In our control structure initial internal ambition levels are so high that no time slack or resource slack is available and the probability of realizing them is low. External ambition levels are communicated with potential customers and therefore the probability of realizing them must be high. These external ambition levels are lower than the internal ambition levels. During the project both internal and external ambition levels can be adapted in accordance with project progress and time remaining until they converge at the deadline. Several work packages are allocated to engineers at the same time to guarantee an optimal work pressure. When the work pressure is too high or when work packages cannot be realized at the planned due dates, project leaders can decide to adjust these due dates or to adjust internal ambition levels. It is suggested that project leaders must try to control the work pressure of engineers at an optimal engineer efficiency. If engineer efficiency is high, the probability that internal ambition levels can be realized increases. But, realizing a high efficiency seems only possible when engineers do not know the difference (slack) between internal and external ambition levels. If they are aware of this slack, they may perceive a lower work pressure resulting in a lower efficiency.

This proposed control structure differs at vital places from traditional project planning and control techniques. First, progress of radical NPD projects is measured by engineers at fixed time intervals. Event-oriented milestones are not suitable in radical NPD projects because of the uncertainty that an event will or can occur. Therefore time-oriented milestones or review points are advocated. Second, progress can be controlled by adjusting planned start and finish dates of work packages. This influences perceived work pressure of engineers and also the sequence in which they work on their work packages (the work package with the earliest due date is processed first). Third, it is suggested that radical NPD projects should be mainly controlled by making use of internal and external ambition levels of functional specifications.

Conclusions and suggestions for further research are given in chapter 8.

Samenvatting

In dit proefschrift analyseren we de beheersing van radicale nieuwe productontwikkelingsprojecten (NPD) die beperkt worden door de tijd (leveringsdatum). We onderzoeken karakteristieken van radicale NPD projecten en van ontwikkelaars die ontwikkelingsprocessen in deze projecten uitvoeren. Bovendien bestuderen we de beheersing van radicale NPD projecten in relatie tot deze karakteristieken. We introduceren ook een beheersingsstructuur voor radicale NPD projecten, die bestaat uit de belangrijkste beslissingsfuncties, de doelen van en de onderlinge relaties tussen deze functies en de posities in een organisatie waaraan deze beslissingsfuncties toegewezen zouden moeten worden.

Hoofdstuk 2 beschrijft de resultaten van een literatuurstudie van NPD projecten, in het bijzonder karakteristieken van radicale NPD projecten en van technieken om deze projecten te plannen en te beheersen. Vanuit een operationeel beheersingsperspectief is technologische onzekerheid geïdentificeerd als een belangrijke eigenschap van radicale NPD projecten. Het beïnvloedt de uitvoering van ontwikkelingsprocessen op verschillende manieren:

- functionele product specificaties staan niet vast aan het begin van een project, maar kunnen gedurende het project veranderen, wat kan leiden tot het toevoegen of verwijderen van werkpakketten en dus tot een variabele werkpakket structuur;
- ongeplande of onverwachte werkpakketten mogen aan de geplande werkpakketten toegevoegd worden, als gedurende de uitvoering van een werkpakket een nieuw probleem ontstaat dat zo omvangrijk is dat een nieuw werkpakket gedefinieerd moet worden;
- bewerkingstijden van werkpakketten zijn moeilijk te voorspellen omdat werkpakketten bestaan uit ontwerpproblemen waarvan de oplossing vaak plotseling gevonden wordt.

Vorig onderzoek naar het beheersen van onzekere situaties onthult dat zowel mechanistische als organische benaderingen vereist zijn om met onzekerheid om te kunnen gaan. Bestaande project plannings- en beheersingstechnieken, zoals PERT en GERT, zijn mechanistisch van aard. Deze traditionele technieken nemen aan dat de werkpakket structuur stabiel is, dat (stochastische verdelingen van) bewerkingstijden van werkpakketten bekend zijn en dat projectresultaten vast staan. Bovendien negeren deze technieken de menselijke aspecten van het uitvoeren van ontwikkelingsprocessen.

In Hoofdstuk 3 veronderstellen we wat zou kunnen gebeuren als radicale NPD projecten gepland en beheerst worden met traditionele, mechanistische technieken. Uit psychologisch onderzoek is bekend dat twee karakteristieken van menselijk gedrag van bijzonder belang zijn met betrekking tot radicale NPD projecten. Ten eerste, mensen hebben de neiging om optimistisch te zijn als ze bewerkingstijden inschatten van werkpakketten die ze moeten uitvoeren. Ten tweede, de efficiency van mensen is afhankelijk van de (waargenomen) werkdruk. De efficiency neemt af als de werkdruk laag of hoog is. Als de werkdruk gemiddelde waarden aanneemt, levert dit de hoogste efficiency op. In dit hoofdstuk geven we aan hoe deze twee karakteristieken radicale NPD projecten beïnvloeden. Als compensatie voor onzekerheid en optimisme in deze NPD projecten, leidt het gebruik van traditionele plannings- en beheersingstechnieken tot projectplannen waarin speling is ingebouwd. Deze speling kan benut worden voor de oplossing van nieuwe problemen die nog niet bekend waren bij de start van het project. Maar, als speling in tijd wordt opgenomen in een projectplan, nemen ontwikkelaars een lagere werkdruk waar met als gevolg een afnemende efficiency. De speling zal verloren gaan, omdat het verbruikt wordt door inefficiënt gedrag. Als dan een nieuw probleem wordt ontdekt en meer (onvoorzien) werk gedaan moet worden, is er weinig of geen speling meer over en de geplande einddatum van het project wordt overschreden. Daarom veronderstellen we dat het gebruik van traditionele project plannings- en beheersingstechnieken voor radicale NPD projecten leidt tot een planningsparadox; de speling in tijd, die in het projectplan is opgenomen om te voorkomen dat het project uitloopt, zal menselijk gedrag veroorzaken dat er juist toe leidt dat het project uitloopt. Het gebruik van deze technieken voor de planning en beheersing van radicale NPD projecten wordt dus betwijfeld.

In Hoofdstuk 4 is een beschrijving gegeven van een grote Europese producent van optische machines in de semi-conductor industrie. In dit bedrijf hebben we empirische studies in radicale NPD projecten uitgevoerd. Eerst moest geverifieerd worden of de ontwikkelingsprojecten in dit bedrijf wel gekarakteriseerd konden worden als radicale NPD projecten. Daarom werden eerst drie kleinschalige empirische studies uitgevoerd, die beschreven zijn in Hoofdstuk 5. Deze studies maakten gebruik van gegevens die al beschikbaar waren (in databestanden) in het bedrijf. Als de projecten gekarakteriseerd konden worden als radicale NPD projecten, dan zou een diepte analyse van deze projecten in het bedrijf gerechtvaardigd zijn. De eerste kleinschalige studie toonde aan dat geplande doorlooptijden van werkpakketten bijna altijd veel korter zijn dan werkelijke doorlooptijden. Dit kan wijzen op optimistische schattingen en/of het ontstaan van nieuwe problemen binnen een werkpakket. In de tweede kleinschalige studie werd een groep, bestaande uit ontwikkelaars en projectleiders, gevraagd om redenen te bedenken waarom radicale NPD projecten moeilijk te plannen zijn. De twee belangrijkste redenen waren: veranderende functionele specificaties (die een instabiele werkpakket structuur veroorzaken) en onjuiste of optimistische schattingen van benodigde capaciteit. In de derde kleinschalige studie werd het aantal werkpakketten waaruit een mechanisch NPD project bestond elke twee weken gemeten. De studie toonde aan dat dit aantal werkpakketten erg vaak veranderde. De werkpakket structuur in dit radicale NPD project leek dus instabiel te zijn. De resultaten van de studies waren in overeenstemming met de karakteristieken van radicale NPD projecten en van menselijk gedrag die in de literatuur gevonden waren. Dit rechtvaardigde een diepte analyse in dit bedrijf.

In Hoofdstuk 6 wordt deze diepte analyse beschreven waarin de volgende veronderstellingen over radicale NPD projecten en menselijke gedrag getoetst werden:

- ongeplande werkpakketten ontstaan gedurende het project;
- ontwikkelaars zijn optimistisch als ze bewerkingstijden moeten schatten;
- een curvi-lineair verband bestaat tussen waargenomen werkdruk van ontwikkelaars en hun efficiency.

De analyse werd uitgevoerd in twee radicale NPD software projecten. Software projecten worden in het bedrijf beschouwd als minst onzeker vergeleken met projecten in andere technische disciplines. Daarom mogen we aannemen dat als in deze projecten ondersteuning voor onze veronderstellingen wordt gevonden, deze ondersteuning ook gevonden zal kunnen worden in andere radicale NPD projecten. Gedurende een periode van 42 weken werden 19 ontwikkelaars van de twee software projecten gevraagd om wekelijks een schatting te maken van het aantal uren dat ze gewerkt hadden aan werkpakketten en van het aantal uren dat nodig was om deze werkpakketten af te ronden. Deze schattingen en de werkelijke bewerkingstijden en doorlooptijden werden gebruikt om de veronderstellingen te toetsen. Voor alle veronderstellingen vonden we ondersteuning in deze twee software projecten. Ten eerste, werkpakket structuren van de projecten veranderden gemiddeld elke twee tot drie weken. Ten tweede, een lineair verband leek te bestaan tussen geschatte en werkelijke bewerkingstijden. Hoewel bewerkingstijden niet zo onzeker waren als we verwachtten, waren werkelijke bewerkingstijden in het algemeen wel langer dan de geschatte waarden, wat aanduidde dat ontwikkelaars wel optimistisch lijken te zijn. Ten slotte, een significant hogere efficiency van ontwikkelaars werd gevonden bij een middelmatige werkdruk, vergeleken met de efficiency behorende bij een relatief lage of hoge werkdruk. Als bijproduct van deze diepte analyse, werd ook informatie verzameld over de manier waarop radicale NPD projecten in dit bedrijf werden beheerst. Deze informatie werd gebruikt bij het ontwerpen van een beheersingsstructuur voor radicale NPD projecten.

Gebaseerd op de empirische studies worden in Hoofdstuk 7 vereisten afgeleid voor het beheersen van radicale NPD projecten. Daarnaast wordt een beheersingsstructuur voor deze projecten geïntroduceerd. Deze beheersingsstructuur helpt bij het oplossen van het radicale NPD beheersingsprobleem: gegeven bepaalde minimale doelen (externe ambitie niveaus) betreffende functionele product specificaties, gegeven de deadline waarop het nieuwe product gereed moet zijn, en gegeven de capaciteit, hoe kunnen we: 1) gewenste (interne) ambitie niveaus van functionele product specificaties vaststellen; 2) werkpakketten voor ontwikkelaars definiëren; 3) ambitie niveaus van werkpakketten en/of werkpakket structuren variëren; 4) werkpakketten aan ontwikkelaars toewijzen. De geïntroduceerde beheersingsstructuur lost de planningsparadox op. Traditionele plannings- en beheersingstechnieken, zoals PERT en GERT, nemen aan dat projectdoelen vast staan en suggereren dat in een projectplan speling in tijd wordt opgenomen, die gebruikt kan worden om ongeplande problemen op te lossen. Deze speling leidt tot een lage waargenomen werkdruk door ontwikkelaars en dus tot een lagere efficiency. In onze beheersingsstructuur worden de interne ambitie niveaus zo hoog gedefinieerd, dat geen speling beschikbaar kan zijn. De kans dat deze interne ambitie niveaus gerealiseerd worden is laag. Externe ambitie niveaus worden aan potentiële klanten gecommuniceerd en daarom moet de kans dat deze gerealiseerd worden hoog zijn. De externe ambitie niveaus zijn lager dan de interne niveaus. Gedurende het project kunnen zowel externe als interne ambitie niveaus aangepast worden in overeenstemming met de voortgang van het project en de resterende tijd, totdat de beide niveaus convergeren als de deadline nabij is. Aan ontwikkelaars worden meerdere werkpakketten tegelijkertijd gealloceerd om een optimale werkdruk te garanderen. Als ontwikkelaars de werkdruk te hoog vinden of als werkpakketten niet op de geplande einddata gerealiseerd kunnen worden, kunnen projectleiders beslissen om deze einddata aan te passen of om de interne ambitie niveaus te veranderen. Het is belangrijk dat projectleiders de werkdruk van ontwikkelaars beheersen, zodat een hoge efficiency wordt gehaald. Immers, als de efficiency hoog is, wordt de kans dat interne ambitie niveaus worden gerealiseerd groter. Maar, het realiseren van een hoge efficiency lijkt alleen mogelijk te zijn als ontwikkelaars het verschil tussen externe en interne ambitie niveaus niet weten. Als ontwikkelaars de speling die hiertussen bestaat kennen, is het mogelijk dat ze een lagere werkdruk waarnemen met als gevolg een lagere efficiency.

De voorgestelde beheersingsstructuur verschilt van traditionele project plannings- en beheersingstechnieken. Ten eerste, de voortgang van radicale NPD projecten wordt op vaste tijden (periodiek) geschat door ontwikkelaars. Mijlpalen die samenhangen met gebeurtenissen zijn niet geschikt voor radicale NPD projecten vanwege de onzekerheid dat deze gebeurtenissen kunnen of zullen plaatsvinden. Daarom worden tijd-georiënteerde mijlpalen of reviews gebruikt. Ten tweede, de voortgang kan beheerst worden door de geplande start- en einddata van werkpakketten aan te passen. Dit beïnvloedt zowel de waargenomen werkdruk van ontwikkelaars, als de volgorde waarin zij werkpakketten uitvoeren (het werkpakket met de meest nabije einddatum wordt het eerst uitgevoerd). Ten derde, radicale NPD projecten zouden vooral beheerst moeten worden door gebruik te maken van interne en externe ambitie niveaus van functionele product specificaties.

Conclusies en aanbevelingen voor toekomstig onderzoek worden gegeven in Hoofdstuk 8.

Curriculum Vitae

Kim van Oorschot was born on September 27, 1973 in Rotterdam. In 1991 she received her VWO diploma from the Guillaume Farel school in Ridderkerk, after which she started studying Industrial Engineering and Management Science at the Technische Universiteit Eindhoven. She received her Master's Degree in 1996 after a research project at Stork Nolte EMT concerning optimal batch sizes in a production plant. In 1997, she started as a Ph.D. student at the Research School for Operations Management and Logistics (Beta), Technische Universiteit Eindhoven, concerning the analysis of radical NPD projects from an operational control perspective. This thesis concludes the research. From September 2001 she will start to lecture at the Technische Universiteit Eindhoven and her research efforts will be positioned in the Research School for Operations Management and Logistics (Beta).

Stellingen

behorende bij het proefschrift

Analyzing Radical NPD Projects from an Operational Control Perspective

van

Kim van Oorschot

I

Clark en Wheelwright pleiten er voor dat ontwikkelaars in *product development projects* over ten hoogste twee taken tegelijk beschikken. Echter, als ook rekening wordt gehouden met de werkdruk van ontwikkelaars zouden zij ook over meer dan twee taken mogen beschikken.

(dit proefschrift, Hoofdstuk 3 en 7 en K.B. Clark en S.C. Wheelwright, 1993, Managing new product and process development: text and cases)

II

In de planning van werkpakketten van *radical NPD projects* dient geen speling in tijd opgenomen te worden, daarentegen is er wel speling tussen externe en interne (of minimale en gewenste) functionele product specificaties nodig.

(dit proefschrift, Hoofdstuk 7)

III

Gegeven de instabiliteit van werkpakketten van een *radical NPD project* is het vanuit beheersingsperspectief niet zinvol om een gedetailleerd netwerk van werkpakketten te maken. Daarentegen is het voor de beheersing van deze projecten wel noodzakelijk om het verschil tussen externe en interne functionele product specificaties te weten.

(dit proefschrift, Hoofdstuk 3 en 7)

IV

Omdat de werkdruk die door ontwikkelaars in *radical NPD projects* waargenomen wordt, lager kan worden als zij op de hoogte zijn van de speling in functionele product specificaties, kunnen projectleiders het bestaan van deze speling beter niet bekend maken.

(dit proefschrift, Hoofdstuk 3 en 7)

V

In bedrijfskunde opleidingen wordt nauwelijks aandacht besteed aan drie kernvaardigheden van een bedrijfskundige: zwijgen, luisteren en observeren.

VI

Een wetenschapper die empirisch onderzoek uitvoert op bedrijfskundig gebied, dient tweetalig "opgevoed" te zijn. Zowel de taal van de wetenschap als de taal van het bedrijfsleven dienen beheerst te worden.

VII

Het onderstaande citaat geldt voor elke rapportage over empirisch onderzoek.

Het is allemaal gebaseerd op waargebeurde feiten, behalve de fictieve gedeelten.

(uit: Herman Brusselmans, 2000, De terugkeer van Bonanza)

VIII

Als een vrouw direct is, is ze voor een man subtiel genoeg.

IX

Om een liefde onvergetelijk te maken, dient men te beschikken over een goed geheugen.

X

Slaap kan je te veel hebben als je het te weinig krijgt.