CHALMERS

Chalmers University of Technology, Sweden

Implementing Product Data Management in Product Development Projects

Master of Science thesis by:

Petter Månsson

Daniel Nyberg

School of Mechanical Engineering
Department of Product and Production Development
Chalmers University of Technology
Göteborg, Sweden, 19-12-2002

ABSTRACT

The goal of engineering education is to provide the students with a high degree of technical knowledge and wide array of personal skills that will allow them to function in real engineering teams. To accomplish these requirements Chalmers together with three other technical universities has developed a program with the purpose to educate the students in how to Conceive-Design-Implement-Operate (CDIO) technical systems. The purpose of this thesis work is to change and develop the Product Development (PD) course, given to master level students at Chalmers, in line with the requirements CDIO and the examiner have set up for the education. As part of this a product data management (PDM) system was implemented in the PD course to better prepare students to perform design activities in a software environment often used in industry.

To get an overall idea of how this could be achieved we started with a wide funnel analyzing the PD process in general and in particular, relevant parts for the PD course. The conclusion from that was that the success in the product development today is very much dependent on how product and project data is structured and the rate of communication. Furthermore, the ambition to shorten the lead times even more makes communication a crucial success factor in the future. Along with that comes an increased need for methods and tools to structure the information. In this thesis work it was predetermined that PDM was the aiding computer tool, thus the funnel became narrower. A severe limitation of current PDM systems is the lack of functionality for requirements management, which was examined.

A stakeholder analysis, with earlier course participants and instructors, was made to get a clear picture of the functionalities required, forming a requirements specification. Since, the utilization of the PDM tool is voluntarily in the course, an important requirement that was found was to not make the system to hard to learn with the risk of scaring the students away with the impression of the PDM system as just another software to learn. However, the overall impression is that the implemented system is easy to learn, and to assure that there will be no trouble and no extra load for the project group a user's manual was created to guide them. The environment in the PDM system was customized according to the requirements specification and the objectives. The designed environment has been verified through a test case, a simulated design project, to assure that everything was in order and to find remaining problems.

The conclusion is that the project has been successful in all objectives except one; the system was up and running at the time the groups needed it, the system and the information in it are accessible from any Internet connected computer encouraging distributed work, image service functionality was investigated and implemented, the project is supported by the budget functionality provided in the PDM system, and manuals will guide the students and help them use the system in the right way. Due to two things this thesis work failed to create requirement management functionality in the PDM system, first there was installation problems and the project was behind time schedule, and the second problem was that special software development tools was needed. The overall judgement of the system, without the possibility to considerate the evaluation at the course end, is that it will be successful.

Keywords: Product development, Product data management, PDM, Student projects

ACKNOWLEDGEMENT

The thesis work has been carried out in the Department of Product and Production Development, Chalmers University of Technology. In the Master of Science in Mechanical Engineering, the thesis work is the final element in the Product Development specialization area. The degree project corresponds to 20 University points.

We would like to acknowledge all the people that have been involved in the work and made it come true. First of all we want to thank our instructor Fredrik Andersson, candidate for the doctoral program at Chalmers University of technology, for the support and understanding he has given us during these months. We also want to acknowledge our examiner professor Johan Malmqvist for the support he has given and the patience he has shown.

Finally we want to acknowledge all the people working at the department, which made the lunches interesting as well as instructive. Now we know that it is bad manner to wear a hat indoors, we are deeply grateful.

Göteborg, December 2002

Petter Månsson

Daniel Nyberg

TABLE OF CONTENTS

APPENDIX

CHAPTER 1. INTRODUCTION

The first chapter explains the background of the thesis work. It also describes the purpose, objectives, method and delimitations. The final part in this chapter is the thesis outline, which will provide a support for the reader.

Background

In contemporary undergraduate engineering education, there is a seemingly irreconcilable tension between two growing needs. On one hand, there is the ever-increasing body of technical knowledge that it is felt that graduating students must command. On the other hand, there is a growing recognition that young engineers must possess a wide array of personal, interpersonal, and system building knowledge and skills that will allow them to function in real engineering teams and to produce real products and systems. [www.cdio.org, 2002-06-01]

To meet these needs Chalmers University of Technology (Chalmers), The Royal Institute of Technology and Linköping University, together with the Massachusetts Institute of Technology, run a project that intends to design and develop a new model of engineering education. Working together, the four universities have developed a new educational approach with the purpose to educate students how to Conceive-Design-Implement-Operate (CDIO) complex, value-added engineering systems, within a modern team-based engineering environment.

One major component in the CDIO's concept is to give the students the opportunity to learn experientially through team based design-built-test projects, both in modern classrooms and in a workshop laboratory. Hence, Chalmers wants to improve the Product development (PD) course through an increased support for the student group, in the product development process, through the implementation of a product data management (PDM) system.

Purpose

The purpose with this thesis work is to change and develop the PD course in line with the requirements CDIO and those responsible for the PD course have set up for the education. Designing an environment in the PDM system and a work procedure that support the communication between the project group members and between the project group and the instructors both at Chalmers and the client companies will do this. This improvement will also make the design process more efficient and most probably the products better than before.

Objectives

- At the start of the course the PDM system should be up and running, i.e. the system should be properly installed and working.
- Investigate the requirements on the PDM system.

- The access to the PDM system should be possible from a remote position. For instance at Chalmers and at home.
- Image service that admits view functionality. For instance 3-D CAD drawings.
- The PDM-system should have requirement management functionality that shows the connections between customer needs, technical requirements, functions and solutions. An example of this is shown in figure 1.1.

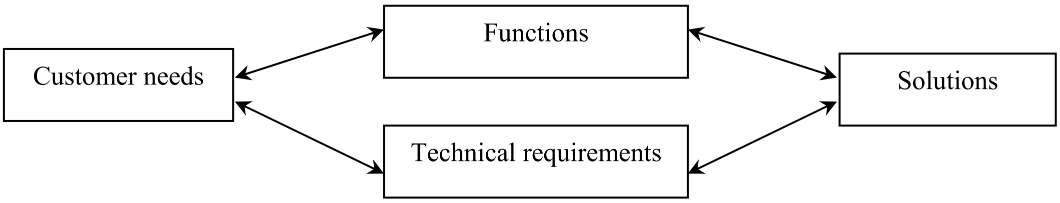
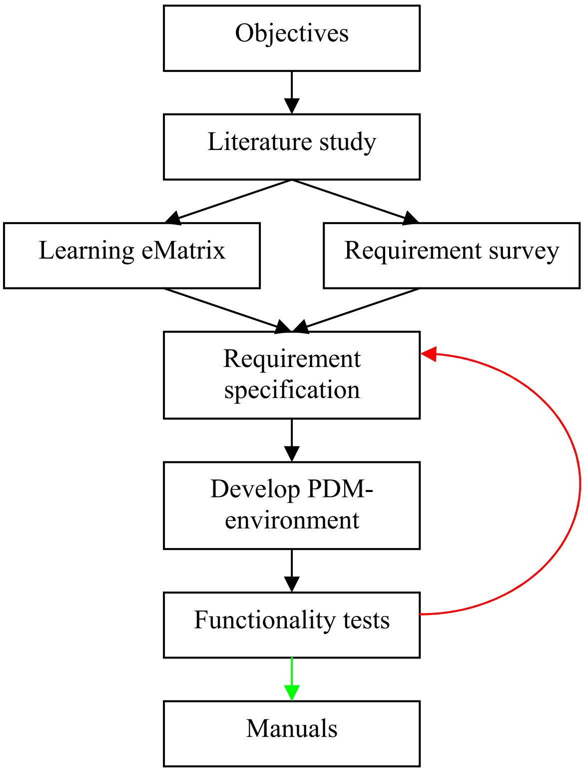


Figure 1.1: Connections between customer needs and solutions.

- Budget functionality.
- Writing a manual that describes how to use the PDM environment in a team based project.

Method

The method started with stating the objectives for the thesis work. To achieve these goals and to understand the subject the next steps are a literature study followed by learning the PDM program and doing a stakeholder survey. On the basis of the knowledge derived from these procedures a requirement specification was stated. Thereafter the PDM environment was developed and tested to assure that the system deliver what is supposed to.



Delimitations

The thesis work does not include an investigation of which PDM-system to use since Chalmers already works with the system eMatrix from MatrixOne. The work focuses on the PD course and its conditions, and does not treat full-scale development projects from idea to production.

Outline of the thesis

Chapter 2, Frame of reference presents an overview of the product development process, PDM, and requirements management. The aim is to understand how to develop a PDM environment to support the project groups in the product development process. To make sure that the PDM environment really can support the project groups in the product development process, these areas must be studied. And since requirements management support in the PDM system was desired, this area also needs to be studied.

Chapter 3, In the Requirements analysis chapter stakeholders and their requirements regarding the PDM system are described. The requirements have been compiled into a specification, which aims to concretise the presentation of the problem and to ensure that all stakeholder requirements have been considered. The specification is drawn up on the basis of requirements and wishes from the assigner to the project and remaining stakeholders. The stakeholders and their most important requirements are described below.

Chapter 4, eMatrix chapter presents the PDM system, eMatrix, used in this thesis work. The aim is to briefly describe the different web applications and their functionality.

Chapter 5, Project support chapter discusses how to, in the best way, give the project group in the PD course support in the product development project. The aim is to make the PDM system as easy to work with as possible, and that includes making a manual and creating helpful templates. The chapter also treats supportive viewing functionality, which is helpful when working in distributed teams.

Chapter 6, Case study chapter a test project is set up and the system is evaluated and discussed from a project member point of view.

The thesis work will end with a conclusion (**chapter 7**) and a recommendation of interesting areas that could be investigated in future work (**chapter 8**).

CHAPTER 2. FRAME OF REFERENCE

This chapter present an overview of the product development process, PDM, and requirements management. The aim is, as said in the introduction, to develop a PDM environment to support the project groups in the product development process. To make sure that the PDM environment really can support the project groups in the product development process, these areas must be studied. And since requirements management support in the PDM system was desired, this area also needs to be studied. Each area will be ended with a state of the art, where an overview of how it is used today, and where the research is heading will be presented.

Product development

A product is something that is made by an enterprise and sold to a customer. Product development process exists of several types for example market pull, technology push, and process intensive, see Ulrich and Eppinger [2000]. This thesis work focuses on the market pull situation, which is the type of product development taught in the PD course. The market pull process begins with a set of activities that starts with perception of a market opportunity and ends with a new product. As late as in the 1980's the product development activity was considered to be a pure design work, done by a designer isolated from the other functions in the enterprise [www.ne.se, 2002-08-21]. After the 80's the product development lead time dramatically increased and the environment became more competitive. To shorten the development cycle companies has gone from a sequential working procedure to a simultaneous and integrated way of working. While the market has become global, intense, and dynamic, the development of new product and processes has more and more turned out to be key factors of succeeding.

Process model

A process is a sequence of steps that transforms a set of inputs to a set of outputs. The process model is used, on an abstract level, to formally describe how the product development process can be carried out in a company. A generic product development processes typically consists of six phases; planning, concept development, system-level design, detail design, testing and refinement, and production ramp-up [Ulrich and Eppinger, 2000], see figure 2.1.

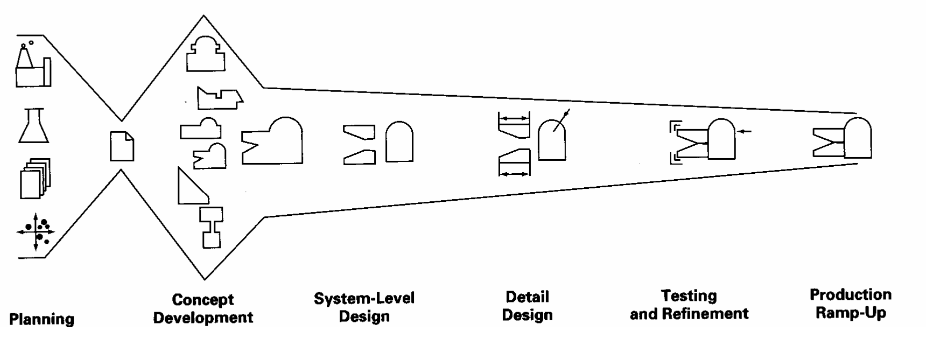


Figure 2.1: The generic product development process [Ulrich and Eppinger, 2000]

Product planning

Companies deal with product planning in different ways, and often it is a manager or a staff members opinion that decides what product ideas to develop and when they should be introduced to the market. However, the situation today has made companies more aware of the importance of planning the product development more accurately. Systematic methods are frequently used to find new product ideas. Several systematic product planning approaches exist but these will not be discussed in detail. Further information about these procedures is available in "Product Design and Development" [Ulrich and Eppinger, 2000], and in "Engineering Design" [Pahl and Beitz, 1996].

The stimuli, from inside or outside the company, are usually identified by the market department. To grab these opportunities, companies have to start a procedure of product planning that starts with an analysis of the situation and continues with formulating search strategies, finding product ideas, select product ideas. The output from the product planning phase is the project mission statement, where the target market for the product, business goals, key assumptions, and constraints are specified.

Concept development

In the concept development phase, see figure 2.2, the customer needs are identified and a target specification is established. On the basis of the specification alternative product concepts are generated and evaluated. One or more of the concepts are then selected for further development and testing, and the target specifications, set earlier in the process, are revised. Finally the team creates a detailed development schedule and identifies the resources required to complete the project. The process also includes economical analysis, benchmarking of competitive products, and modelling and prototyping.

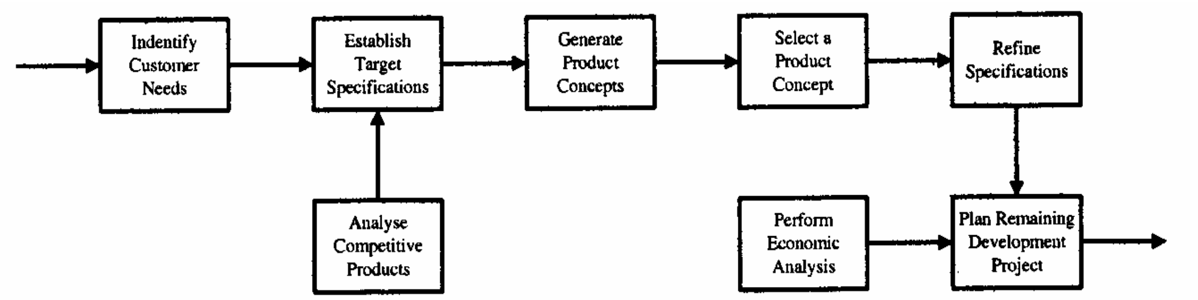


Figure 2.2: Steps in the concept development phase [Ulrich and Eppinger, 2000]

Rarely does the entire process proceed in purely sequential fashion, completing each step before beginning the next. In practice the steps may be overlapped in time and iterations is often necessary to get the optimum result.

System level design

The purpose of the system-level or product architecture phase is to define the basic physical building blocks of the product in terms of what they do and what their interface are to the rest of the device. In other words, a product can be thought of in both functional, in terms of the individual operations and transformations that contribute to the overall performance, as well as physical terms, which are the parts, components, and subassemblies that realises the products functions. The functional elements of a product are the individual operations and transformations that contribute to the overall performance of the product, and the physical elements of a product are the parts, components, and subassemblies that in the end realize the products functions.

Detail design

The detail design phase includes the complete specification of the geometry, materials, and tolerances of all of the unique parts in the product, and the identification of all of the standard parts to bee purchased from suppliers. In this phase, the tools are also developed as well as the production system.

Testing and refinement

The testing and refinement phase involves the construction and evaluation of multiple preproduction versions of the product. Tests and evaluations of models and prototypes are continuously made during the entire development process not only in this phase, although it is a difference in the quality of the prototypes. More information about the differences is discussed in "Product Design and Development" [Ulrich and Eppinger, 2000].

Production ramp-up

In the product ramp-up phase is the product made by using the intended production system. The purpose of this phase is to train the workers and to work out any remaining production problems.

The Product Development course

The Product Development course aims to train the students in real industrial projects, with budget and work in cross-functional groups. The course is also suited to, with some changes, fit into the program from a CDIO perspective, to give a good understanding of how to conceive, design, implement, and operate. As the course only lasts for about eight months, the product development process has to differ slightly from the one described earlier and emphasize the middle of the process, since a part of the planning phase already has been done by the company that formulated the project assignment. The output from the project will not, for natural reasons, reach to the production ramp up but will end up with a prototype that is analysed and evaluated. The development process will be governed by a stage gate process where a progress report is presented at each gate, see figure 2.3.

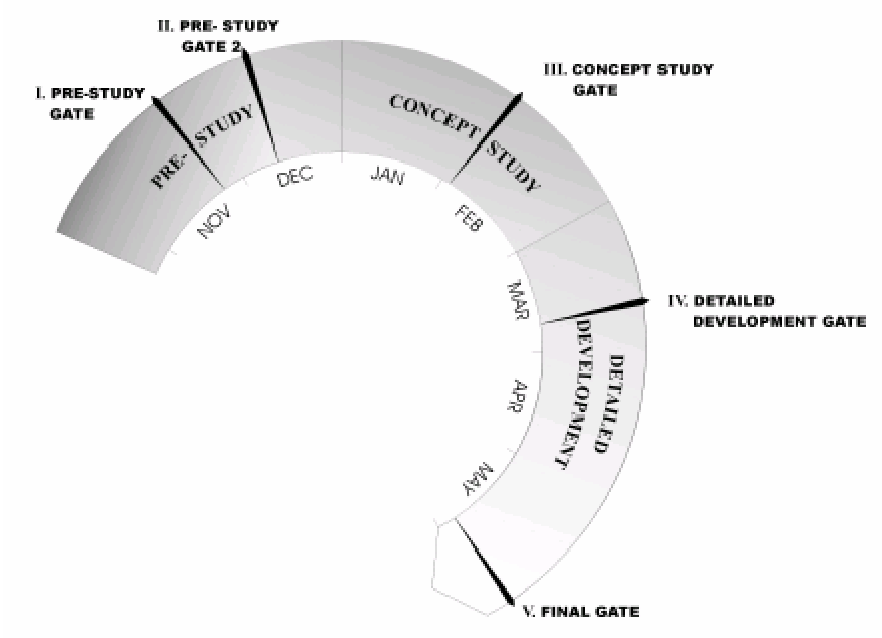


Figure 2.3: Product development process in PD course

The process is divided in to three main phases, these are: the pre-study phase, the concept study phase, and the detailed development phase. In the pre-study phase the group meets the client company, understand their assignment and translate it into at problem definition. They also establish a project plan and a preliminary budget. The result from the pre-study phase is a requirement specification based on the stakeholder analysis that been made. This phase is followed by the concept study phase, were the requirement specification is the foundation for the brainstorming and concept generating activities. The concept ideas are then evaluated to ensure that the best concept is selected for further development. The group also has to revise the problem definition, project plan, and the budget on basis of the changes in the project and new knowledge that been captured. In the third and final phase, the detailed development phase, the group makes a detailed study of the product that ends up with a physical prototype to present at the final presentation. There should also be a commercial evaluation and a market launch plan presented. The entire project is also documented in a report.

The course also aims to give knowledge in how to work in integrated and cross-functional teams. To achieve this teams are made up of students from mechanical engineering, industrial engineering, and industrial design. The easiest way to communicate in small project teams like the one in the PD course is maybe not to implement a PDM system. However, to prepare the students of how to work in large scale industrial projects, with several project members and very complex products, a PDM system is needed to keep all product related information structured and available to whom it may concern.

Another issue that differs slightly from industrial projects is the project organization. The project in the PD course has a rather flat organization, see figure 2.4, with instructors guiding the project team when they need support. The team members have different backgrounds and also different special capabilities and therefore also different roles. The most demanding role is probably the project leader role, and it should preferably alter between the project members.

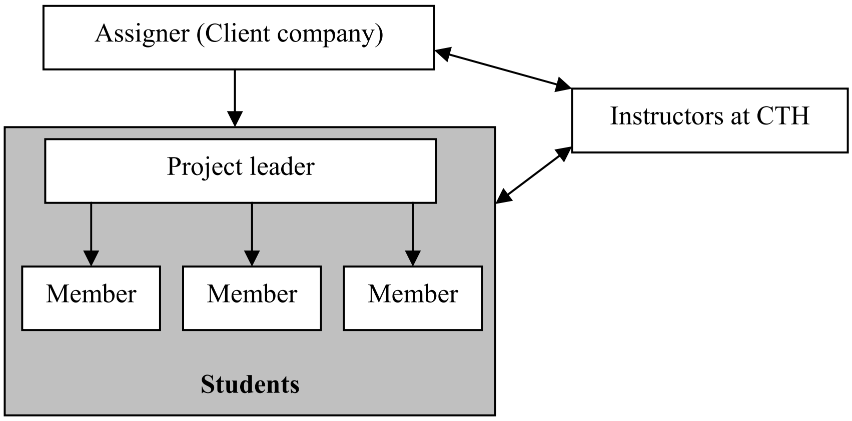


Figure2.4: The project organization in the PD course.

State of the art

Swedish industry finds a great interest in global product development trends. This is the main reason why the Swedish Engineering Design Research and Education Agenda (ENDREA) have done a study with the title “Global Study on Product Development” [Svensén et al, 2000]. The report, which is based on several interviews of market leading large and middle-sized companies, found that the product development processes vary depending on the category and size of the company and project. In general, large companies have a well-defined development process. One of the reasons is the large amount of information that needs to be shared and organized, another is the need of a standard proceeding of the development cycle. In small companies, the process is usually more informal and the information can be stored manually. According to Nilsson et al. [2000] were the most common reasons to develop and implement a product development method to:

- Facilitate coordination, planning and project management,
- Locate where improvements are needed,
- Establish a comprehensive view on an operation level,
- Support knowledge transfer, to oblige customer demands, and
- Show “the outside world” how the business is performed.

One of the areas that advances most rapidly is the collaboration with suppliers and partners in the early phases of the product development process. This need has become more and more important as outsourcing has become an ordinary way of dealing with product development of complex products.

Nilsson et al. [2000] stress the importance of information as one of the key factors for success. Furthermore, the ambition to shorten the lead times even more make the communication a crucial success factor in the future and along with that comes an increased need of methods and tools to structure the information.

Product data management - PDM

PDM, in some literature called cPDm (Collaborative Product Definition Management), is a tool that can be used to manage all product-related information as well as the processes utilized throughout the product's entire life cycle. This product-related information includes geometry, engineering drawings, project plans, part files, assembly diagrams, product specifications, bills of material, and many others. PDM manages product data throughout the enterprise, ensuring that the right information is available for the right person at the right time and in the right form, see figure 2.5. By using PDM in the right way the sharing of information between group members will be improved.

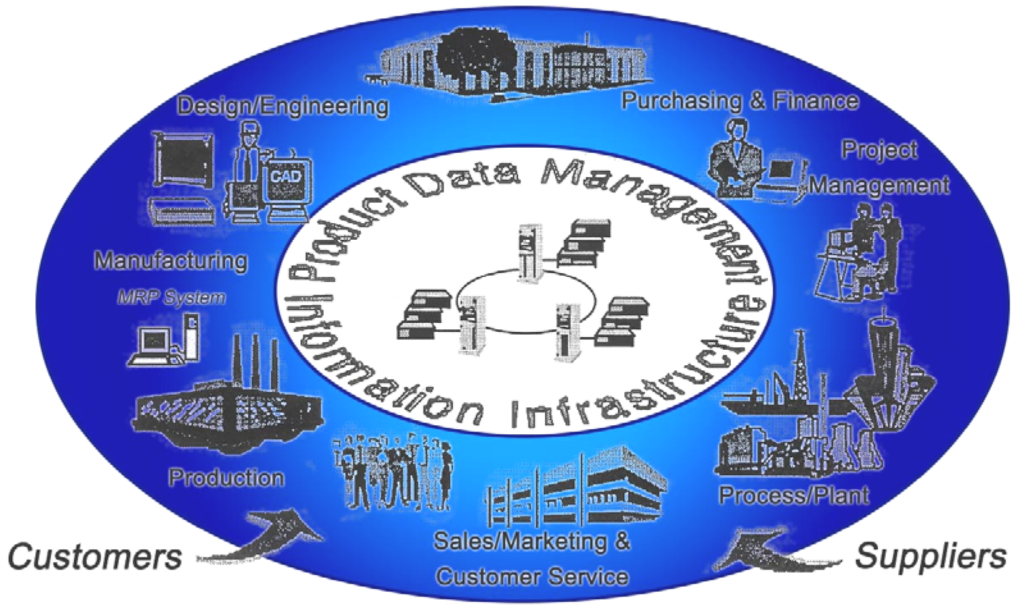


Figure 2.5: PDM throughout organizations [CIMdata, 1998]

PDM function and features

To support any particular type of product development the PDM systems need a basic set of functions. These functions serve all of the disciplines that benefit from PDM. The major components include a set of foundation technologies that support a set of core functions that in turn, support applications and focused business solutions. The model, see figure 2.6, and the definition that follows are based on "collaborative Product Definition management (cPDm): An Overview" [CIMdata, 2001].

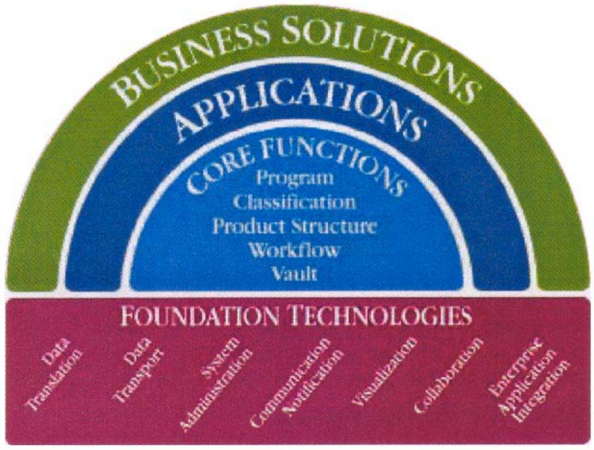


Figure 2.6: The world-class PDM business solution model [CIMdata, 2001]

Foundation technologies

Utility functions provide support that facilitate the use of the system and provide support to the user functions. Utility functions include:

Communication and notification – Users of PDM systems can automatically be notified of critical events concerning the current state of the project or product. E-mail is used to notify people about important events or required actions. PDM minimizes the delays caused by misplaced communication. A functionality called trigger is used spawn notifications and other actions automatically.

Data transport – Users do not need to know where the data is stored while the system keeps tracks of the data location and allows users to access it knowing only the name of data.

Data translation – Data translators can be pre-defined to convert data between different applications and to formats for various display and output devices. These data translations can be done automatically by triggers.

Visualization – Image viewing allows users to view drawings and other design data without need of expensive application software. Image service enhances productivity by facilitating the exchange of information and comments among users.

Collaboration – Collaboration, between different units across geographically distributed sites, becomes possible with the many technologies employed in PDM systems. Examples of technologies are e-mail, audio- and teleconferencing, and collaborative visualization tools. These technologies are designed to support distributed teams to synchronously work together without the need to travel.

Enterprise Application Integration (EAI) – EAI allows information and processes to be shared with other enterprise applications and includes technologies that enable business processes and data to communicate to one another across applications and networks within an extended enterprise.

System administration – The administrator sets up the operational parameters of the PDM system and monitors its performance. Administrative functions include access and change permissions, authorizations, approval procedures, data back-up and security, and data archive.

Core functions

User functions provide the users interface to the PDM systems capabilities including data storage, retrieval and management. PDM systems support five basic user functions. These are:

Data vault and document management – These functions provides secure storage and retrieval of product definition information. An electronic vault is used as a repository to control all kinds of product information. The vault is a data-store that contains some data within itself and controls other externally generated data by managing access to it. Two types of data are stored:

- Product data generated in various applications and
- Meta-data, which is data about PDM controlled information. Typical information that is stored as metadata are part number, date, revision and file location

The data vault also supports Check-in and check-out functions to provide secure storage and access control. To ensure that documents achieve the right status in its life cycle various pre-defined approval processes can be used. The document management contains meta-data that stores information about product data so that changes, release levels, approval authorizations and other data controls can be tracked and audited.

Product structure management (Configuration Management) – Product structure management (PSM) facilitates the creation and management of product configurations and Bill of Materials (BOM). The PDM system tracks configuration changes of versions, effectivities and design variations. It allows user and applications to link product definition data to parts and product structures, which allows users to easily determine which information that will be affected by changes. PSM provides customised views of product information for different users. PSM also support the transfer of product structure and other data between the PDM and enterprise resource planning (ERP) systems.

Workflow and process management – The PDM system can interact with people that work with product data according to the predefined processes of an enterprise to achieve corporate objectives. Repetitive workflows and processes can be programmed as part of a PDM system to route data and work packages automatically, to control and monitor processes, and to provide management reporting. The PDM system also controls the process and procedures that manage how changes are proposed, reviewed, and approved and incorporated into a product and its associated data items. Such a workflow is common in most enterprises, but other workflows exist for design release management, bid preparation, engineering reviews, purchasing, problem tracking and resolution, and contracts management.

Parts management (Classification) – Classification of parts allows similar or standard parts, processes, and other design information to be grouped by common attributes and retrieved for use in products. This lead to greater product standardization, reduced redesign, savings in purchasing and fabrication, and less reinvention of the wheel.

Program management – Program management coordinates the framework of the project that deliver product to market. It provides work breakdown structures (WBS) and allows resource scheduling and project tracking. Program management also provide the ability to relate WBS tasks to the PDM systems knowledge of approval and product configurations. Tasks that are required to complete the project are ordered within work breakdown structures, which can also be groped into hierarchical structures of dependencies. This provides a convenient way to allocate resources and track the projects progress. When the project advances from task to task, actual used resources are recorded against the plan. Completion of activities for each task is tracked and reported through the approval process. According to Mesihovic et al. [2001] program management should be completed with project management information system, which is better suited for tasks as for instance visualizing and reports report activities.

PDM and the Internet

To meet the demands on product development, as mentioned in the product development chapter, companies have to build superior products and deliver it to market faster. Companies must increasingly collaborate within and outside their organization. This is possible with the existing html/Java-based web user interface used in most PDM systems today. The systems enable the users to work in groups even if they are in separate locations and sometimes in many different countries. According to Mendel [1999] the distinct advantage of the client/server model is that it puts application power on the desktop, where it runs at the convenience of the user rather than of the mainframe scheduler.

According to Pramås [2001] are many companies in the process of changing their old in-house developed PDM system and replace it with a commercially available one. Further on he says that distributed PDM systems exist in some companies, but that it is not common that external partners are connected to the system.

The added danger with the Internet is that information access is provided across a wide area, unprotected by the security firewalls most companies have in place to isolate their systems from the outside world. The primary security mechanism currently used on PDM systems is the use of password that identify users, telling the system what each person can see or change. Another drawback with information that is transmitted across the Internet, around any number of routes before being received by a system with the appropriate address, is that unencrypted data becomes visible and vulnerable to those who know how to retrieve it. [Mendel, 1999]

Benefits and drawbacks

The information needed throughout a products life can be managed by a PDM system, making correct data accessible to all people and the systems that have a need to use them. Implementing PDM-systems can benefit the entire business – design and engineering, production operations, purchasing, market and sale. While this work focus on product design and engineering, only benefits regarding this domain will be considered.

According to Philpotts [1998] logical management of product structure information is the key facility in the PDM systems. Using PDM designers and engineers can obtain comprehensive information either via the traditionally part classification route or by interacting with graphical representation of the product structure to find key data. Another benefit is that PDM workflows accelerate the review and approval of change packages, resource allocation and monitoring, new design, design alternatives exploration and process capability evaluation. By monitoring parts and documents status level, PDM facilitates rapid change early in the design cycle, but ensures security and control when design stabilizes. Implementing PDM allows smooth data flow between design and other involved departments.

There are many arguments not to implement PDM [Stark, 2000], for instance: “We have other priorities”, “PDM is too difficult” and “We're engineers. We have to design products, not play around with computers”. Most of the negative aspects of implementing PDM is closely connected to the problems related to changes in the existing organisation.

State of the art

There are a lot of positive effects of implementing a PDM system into a business. But there are also reasons not to or to wait, most of the systems are expensive to buy and it is also a large process to implement the system, which is very resource consuming. These negative aspects have forced the PDM companies to develop new “out of the box” solutions. These, “best practice”, solutions fit to a wide range of companies and their needs. There are three obvious advantages to this type of system: they can be installed quickly in an enterprise, they are cost effective, and they are a good way of getting started [www.technia.com, 2002-09-04].

One area, were some progress has been made, is the collaboration within the company as well as with suppliers in the extended enterprise and lots of effort are put to make PDM a enterprise-wide system. This need has arisen from the fact that products of today are very complex and often are they developed by teams of companies, with different core competencies, rather than by single firms. Another vision is to integrate PDM and ERP (enterprise resource planning) systems, and then eliminate the redundant data and get rid of the lead time for moving data from one system to the other and vice versa.

Finally it should be pointed out that what was said in the introduction, that PDM enable to manage all product-related information throughout the product’s entire life cycle, is not fully true. According to Malmqvist [2001], it is well known that one of the most severe limitations of current PDM systems, towards a “total” digital representation of the product, is the lack of functionality for requirements management.

Requirements management

According to Malmqvist [2001], requirements management has a major impact of whether a project will be successful or not. Two major reasons for failure are an incomplete requirement specification and misunderstood customer needs. Requirements are descriptions of how a system should behave, or of a system property or attribute. Requirement should always be statements of what a system should do rather than a statement of how it should do it. They may be a constraint on the development process of the system [Sommerville and Sawyer, 1997]. Requirements management aim to, for example, make the problem formulation more concrete, make it easier to govern the development process, give the development team a homogeneous view, see Malmqvist [2001]. It is also important to ensure that customer needs, stakeholders, all life cycle phases and prerequisite have been taken into consideration during the entire product development process.

The last ten years have seen an increasing concentration of the rewards from advanced technology. For many types of products, the concept of a "local" market is disappearing. Nowadays companies increasingly work in distributed teams [Stevens et al, 1998]. To be competitive requires highly efficient engineering, design and development activities. If a company wants to succeed on a market they have to be fast and deliver the right product before the competitors. This often means that they have to increase the functionality of the product and still have a fast development process. Requirement management is one key technology to manage this complexity. These processes are normally guided by methods that support handling of requirement.

Requirements management process

Requirements management is a subset of systems engineering, which is about creating effective solutions to problems, and managing the technical complexity of the resulting development. At the outset, it is a creative activity, defining the requirements and the product to be built. Then the emphasis switches, to integration and verification, before delivering the system to the customer. One model of the systems engineering processes is illustrated in figure 2.7.

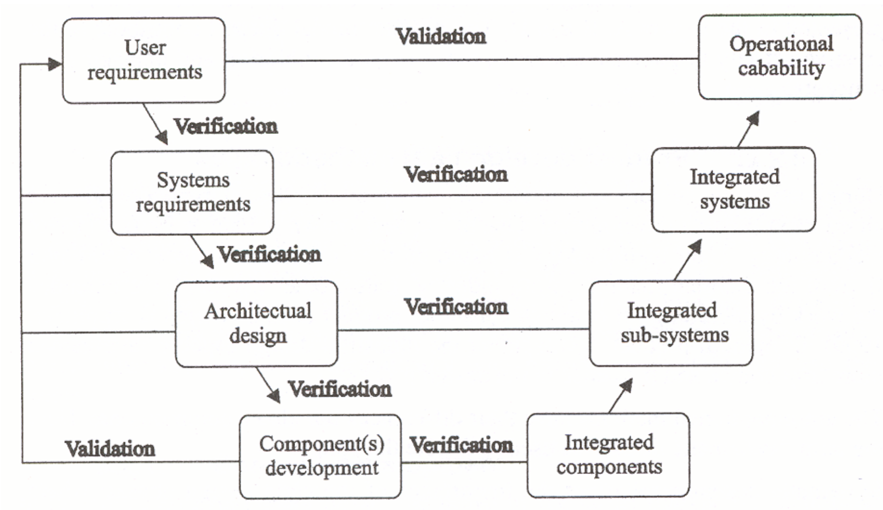


Figure 2.7: The systems engineering process [Andersson, 2001, adapted Stevens et al. 1998]

Requirements management is the process of managing changes to a system's requirements. Requirements evolve because of changes in a system's environment and as customers develop a better understanding of their real needs [Kotonya & Sommerville, 1998]. Requirements process includes all steps from elicitation of requirements to traceability in the product structure. The requirements process is divided into two parts were elicitation to requirements specification is the first, called requirements engineering (RE) process, and the second part, called the system requirements process, includes the modelling of the system and the coupling between requirements specification and the system. The requirements management process is an iterative process where the requirement specification is a dynamic document. The goal is to produce the best cost-benefit ratio with the available recourses. However, the requirements always change with time, which is why traceability to the requirements must be retained.

Requirements engineering process

There are many possible ways to organise requirements engineering processes and they do not transfer well from one organisation to another [Sommerville and Sawyer, 1997]. However, a good requirement engineering process normally includes a set of activities. The process starts with eliciting requirements from the input sources. Examples of these sources are; existing systems information, stakeholder needs, see figure 2.8, which is a modification of the requirements management-process by Kotonya & Sommerville [1998]. The process continues with requirements analysis and negotiation, were the requirements are evaluated in detail. This is necessary because there are inevitably conflicts between the requirements from different sources. The requirement management process ends with a requirements validation, which is a careful check of the requirements for consistency and completeness. The requirement specification can then be completed.

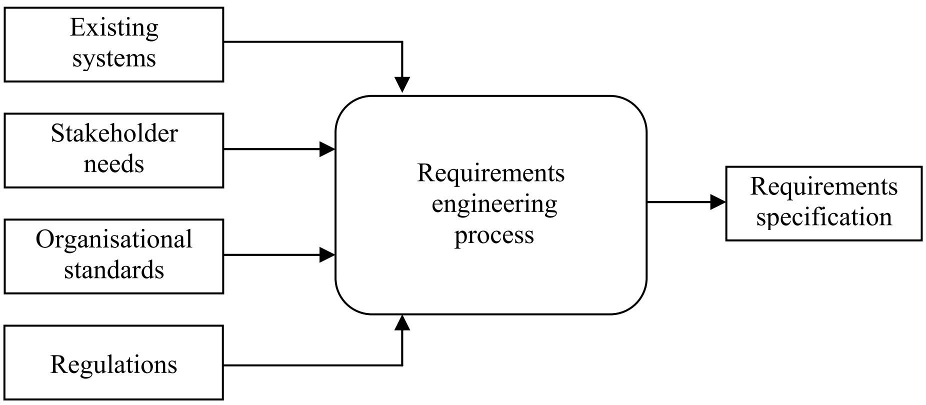


Figure 2.8: In- & outputs of the requirements engineering process [based on Kotonya et al. 1998]

System requirements process

The requirement specification is the foundation for proceeding into step two in the requirements management process, the system requirements process. In this process requirements from the requirements specification are coupled to different objects in the function structure. In the function structure, the main function is decomposed into sub-functions, based on the functional and non-functional requirements from the requirements specification, which in each level describes the total system and ends with a conceptual layout. The coupling between the requirements and the functions result in traceability, which makes it possible to see which requirements to accept, reject, or postpone.

Enhanced function-mean tree

Requirements management is often guided by a requirement method, which is a systematic approach to documenting and analysing system requirements. In this work will an approach adopted from the chromosome model be used, namely the enhanced function-mean tree.

The modelling procedure starts with formulation of the overall functional requirement and constraints on the highest hierarchical level. These formulations can be complex and are most likely derived from some kind of project assignment [Schachinger and Johannesson, 2000]. This overall functional requirement is solved by a design parameter that in turn is decomposed into new requirements at a lower level. This procedure goes on until it reaches the lowest hierarchical level, were the design parameters are materialised by a part structures.

According to Andersson et al. [2000], the enhanced function-mean tree can be the core structure of an object-oriented requirement and concept model. It is a systematic design method that couples the functional requirements and the product structure. The model consists of three different kinds of objects: functional requirements (FR), means (DP) and constraints (C).

FRs are defined as what a product, or element of a product, actively or passively does in order to contribute to a certain purpose. The FR is specified further with attributes stating functional related properties. DPs are physical or an abstract entities, such as software, chosen to fulfil a specific FR, i.e. the organs ("function carriers"), sub-system, components, features that constitute the design solution. Means can be described with attributes, documents and models describing the characteristics of design solution. Cs are in contrast to FR, non-functional requirements, here referred to as constraints, that do not have a specific solution but are instead fulfilled by functional driven means. They bound and add value to the solution space, such as weight, cost, reliability, safety, ergonomic etc.

Objects only make sense in relation to each other. Johannesson [1997] has identified six different relations that are key factors in function-means decomposition and functional coupling analysis, see figure 2.9. These are defined as follows:

A FR ***is_solved_by (isb)*** a DP.
A DP ***is_constrained_by (icb)*** a C.
A DP ***requires_function (rf)*** FRs on the next lower hierarchical level.
A C ***is_partly_met_by (ipmb)*** DPs on the next lower hierarchical level.
Parallel DPs with the same "parent" ***interacts_with (iw)*** each other.
The fulfilment of a FR ***is_influenced_by (iib)*** the choice of a parallel solution (DP).

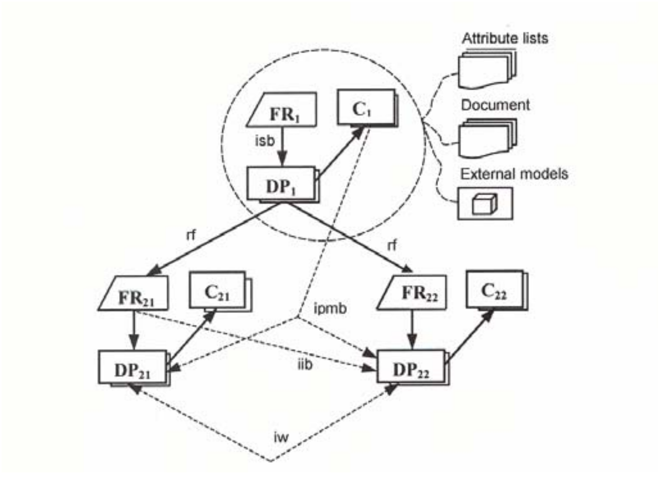


Figure 2.9: Enhanced function-means tree [Andersson, et al., 2000]

State of the art

Recent years requirements management has been given more and more attention and the trends are similar to the ones in PDM. Some requirements management tools are available on the market, for instance DOORS from Telelogic, Slate from EDS, and CORE from Vitech [www.incose.org, 2002-12-09]. Solutions, like enterprise-wide systems, are developed to make the requirements visible to whom it may concern. Another issue is the integration in other systems, i.e. PDM systems. A lot of functions that were visions yesterday are reality in today's commercial requirements management tools. Methods to manage requirement have been developed and improved for some time, and according to Davis and Hickey [2002] has research and development reached to a point were one has to stop and ask oneself a couple of questions. Is it possible to push the technology further? Has this been done before? Is this what stakeholders want?

Concluding the frame of reference

The product development process is a sequence of steps that transforms inputs to outputs. In the PD-course the middle of the process is emphasized. The course project is divided into three main phases; pre-study phase, concept-study phase, and detailed development phase.

The PD-course aims to train the students in real industrial projects, with budget and to work in cross functional groups. To support the project groups, and to prepare the student of how to work in large scale industrial projects it is desirable to implement a PDM system. Today the PDM system vendor develop "out-of-the-box" web applications, this has some major advantages. They can be installed fast, they are cost effective, and it is possible to work from anywhere as long as an Internet connection is available.

A PDM-system is of course well suited for this kind of projects but it is not the whole key, requirements management tool is also desirable. Requirements management has a major impact of whether a project will succeed or not. And it is also important to ensure that customer needs, in all life cycle phases and prerequisite have been taken into consideration during the entire product development process.

The development of the course, with a PDM-system and a requirements management tool will give the project groups a good support in the product development process and will make the collaboration more efficient and finally end up with better products.

CHAPTER 3. REQUIREMENTS ANALYSIS

In the following chapter the stakeholders and their requirements regarding the PDM system are described. The requirements have been compiled into a requirements specification, which aims to concretise the presentation of the problem and to ensure that all stakeholders requirements have been considered. The specification is drawn up on the basis of demands and wishes from the assigner to the project and remaining stakeholders. The stakeholders and their most important requirements are described below.

Stakeholders

To investigate what kind of functionality the stakeholders wants in a PDM system, a qualitative investigation was made. A dialogue with the respondents was done in order to find out how a PDM system could support the group in the product development process. Problems that student groups had worked with previous years were also discussed, this to find possible areas where improvements could be made. The complete requirements specification is to be found in table 3.1.

There are four main stakeholder categories: students, thesis work assigner, instructors at Chalmers, and client instructors. Some requirements are group specific, and are listed under the stakeholder groups, others are common for all groups. The common requirements are listed below.

Communication: In line with what was said in the frame of references, is it well known that effective communication is one of the main success factors when developing new products. The possibility to communicate, student to student, instructor at Chalmers to student, client instructor to student and vice versa, with today's computer tools is no longer only a wish but has become a basic functionality in a PDM system.

Remote login: The ability to access the information in the PDM system should not be restricted to a specific computer or location. One should be able to work from home or any other location if necessary. Another reason why this is important is the fact that the groups are cross-functional and in need of different software while working, hence they have to perform their work from different locations.

Viewer functionality: All stakeholders should to be able to view, for example CAD drawings and other graphically presented information without need of specific software.

Students

The student group comprises all students that are involved in the product development course included, hence are both students from mechanical and industrial engineering as well as industrial design students treated in the same group without making a different between their special competencies and needs.

File management: The system has to be able to handle many kinds of files in a structured way. It has to be possible to store and change files, as the project advances. The system should also be able to handle revisions.

Project plan: The system should support the creation and reviewing of a project plan. i.e. tasks that should be done, duration of tasks, status feedback are some features that should be included in the system.

Budget support: Students have a strong whish to have a tool supporting their budget work. They want the system to support them with the cost plan, actual costs, incurred an easy budget application process and they also want the information to be presented in a way that is easy to understand.

Product structure: The possibility to build product structures and to tie product information to the right level in the product structure is a basic functionality in a PDM system, hence this functionality also has to exist in the PDM system for the product development course.

Meeting functionality: To have a regular meeting procedure where it is possible to leave a message if one could not come to the meeting. Another opinion is that it would be nice to be able to attach an agenda for the meeting where everybody could see what is the main subject today.

Thesis work assigner

This group includes people involved in the development of PD course and the CDIO project at the department of Product and Production Development. They have seen the need and possibilities of a computer support tool in the product development process. Their intention with the PDM system is to make a small pilot project for evaluation and further development in line with the research areas in the department. One of these areas is requirement managements in the product development process and this subject is also one of their requirements on the system.

Requirements management: It is well known that PDM system lack requirements management functionality. This knowledge together with the obvious need of handling requirements, as was described in the frame of reference, raises the need of implementing requirement management in the PDM system.

Activity check-up: To be able to evaluate the pilot system it is desirable to check the activity in the system. It is of great interest to know when and how often the users have accessed the system.

Instructors at Chalmers

The instructor group includes all instructors that in one way or the other have something to do with the overall management of the project. The instructors role is a supporting function for the project group to lean to when the project is in some kind of downturn. The differences in the expertise of the instructors have not been regarded while interviewing the respondents.

Review budget: Student should be able to send a budget application to the project instructor for her/his approval. The instructor at Chalmers should be able to approve or reject the proposal and also be able to add a comment. Instructors at Chalmers should also be able to, in an easy way, grasp the whole budget for the project and see in what direction the project is going.

Progress check-up: The instructors should be able to view the project material with the purpose to check if the project advances according to the project plan.

Client instructors

Client instructors have the same function as the instructors at Chalmers with the difference in that they represent the company for which the project is done.

Security: The ability to access the information from any Internet connected computer should not have an impact of the security of the information stored in the system. This is a very important issue for companies, since student projects can treat sensitive information on companies products, i.e. requirement levels, product descriptions etc.

Table 3.1: Requirements on the PDM system

Stakeholders	Requirements	Functionality	R/W	Weight
Students	Communication	Send/receive mail to a person or group	R	
	Remote login	Access to the system from any Internet connected computer	W	5
	File management	Check in/out functionality	R	
	Project plan	Establish project plan	R	
	Budget support	Send applications, Receive approval, See budget status	W	4
	Product structure	Build digital product structure	W	3
	Meeting functionality	Inform about meeting	W	4
	User manuals	Full description of how to use the system	R	
Thesis work assigner	Requirements management	Manage requirements	W	3
	Activity check-up	Time, Occasions	W	2
	Product structure	Build digital product structure	W	3
Instructors at Chalmers	Communication	Send/receive mail to a person or group	R	
	Review budget application	Accept/reject functionality	W	4
	Progress check-up	Access to project space	W	3
Client instructors	Communication	Send/receive mail to a person or group	R	
	Progress check-up	Access to project space	W	3
	Security	Secure login	R	

W= Wish R=Requirements

CHAPTER 4. EMATRIX

This chapter presents the PDM system, eMatrix, used in this thesis work. The aim is to briefly describe the different web applications and their functionality. Most of the information in this chapter is gathered from product information sheets [MatrixOne, 2002].

Introduction

eMatrix is a PDM system from MatrixOne inc. In Sweden their biggest re-seller is Technia AB, and they sell program licences and services to customers in Finland, Sweden and Norway. General Electrics, Nokia, Esab and Ericsson are some of Technia's customers.

As was mentioned in the frame of reference, PDM systems evolves to be more wide enterprise wide support systems, where the work more highly is distributed and made by everyone in the enterprise. MatrixOne's development of eMatrix is no exception, the eMatrix application of today has a web-based interface, where users typically access the system through an ordinary web browser, for instance Internet Explorer or Netscape.

MatrixOne Value Chain Portfolio

The portfolio includes eight applications, called Centrals. They give, according to Technia, extended possibilities for collaboration within the extended enterprise. The centrals in some cases overlap each other, but they have different special areas. One central specialises in collaboration with suppliers, Supplier Central, another specialises in ad-hoc collaboration within distributed teams, Team Central, and so on. In total are there eight different centrals, but at Chalmers only three of them are installed, and they are:

- Engineering Central,
- Program Central, and
- Team Central

Engineering Central

Engineering Central is an application that provides the ability to electronically define, edit, manage, distribute, and view product engineering data and processes across the enterprise and extended enterprise. Engineering Central manages product engineering data and processes throughout the entire product lifecycle, from concept to retirement. Users can work with part information, bills of material and product structures, associated design and manufacturing documentation, engineering change requests, and engineering change order (ECO) processes. During the concept phase of development, integrations to popular CAD tools automate the capture of mechanical and electronic product structure. During the production phase of the product life-cycle, integrations to most popular enterprise resource planning (ERP) systems facilitate the seamless transfer of product engineering information.

Program Central

Program Central is an application that helps companies manage complex projects that depend on extensive collaboration across global value chains of employees, customers, suppliers, and partners. This application provides management with-real time visibility of a project's status via process, costs, and benefits project dashboards, ensuring that high benefit projects stay on track. To deliver the right information to the right people at the right time, and to reduce the time and cost associated with searching for information, MatrixOne Program Central facilitates the creation of fully supported projects, new product introduction, and information technology.

Team Central

Team Central is an application that offers a virtual workplace for ad-hoc design collaboration with cross-enterprise teams of employees, partners, suppliers, and customers who can work across multiple firewalls to innovate and solve business issues. In today's environment of outsourced product design and manufacturing, global workplaces, compressed product lifecycles, and custom product configurations, virtual teams are a requirement. In the Team Central, you can create a collaborative digital workplace for global team members, called a "WorkSpace." A WorkSpace can be used by geographically distributed team members who offer expertise from disciplines such as design, engineering, manufacturing, procurement, and finance. Within this WorkSpace, members can create folders; organize, view, review, markup, and approve information such as documents and objects; subscribe to event notifications; schedule and access real-time meetings; and use online discussions to find solutions to business issues.

CHAPTER 5. PROJECT SUPPORT

This chapter discusses how to, in the best way, give the project group in the PD-course support in the product development project. The aim is to make the PDM system as easy to work with as possible, and that includes making a manual and creating helpful templates. The chapter also treat supportive viewing functionality, which is helpful when working distributed. The manual is available in appendix I.

The PDM environment

The foundation of the system is, as said in the previous chapter, the eMatrix web application. This system is an out of the box application developed to fit industrial companies and needs in this particularly case some modification and simplification. The system needs to be simple and easy to learn, since the resources and time are limited, there is no time for the two or three days training that it usually takes to learn a program like this.

There are, as said in the previous chapter, three different centrals to choose from. Two of them are intended to be used in development projects, Team Central is intended for ad-hoc development and Program Central is intended for more structured projects. The choice of the Program Central was therefore rather obvious, since the intention is to teach the students how to work in a structured fashion in product development projects.

Before the PDM application was properly installed, an investigation of how to build up the system in the best way was carried out. To make the system easier to understand and overview, without using real names, two project groups and instructors was simulated. Each group consisted of three students, one instructor from Chalmers and one client instructor, see table 5.1.

Table 5.1: The simulated project groups, with students and instructors.

Group Alfa	Group Beta
Adam (student)	Dag (student)
Bert (student)	Eva (student)
Cicci (student)	Fia (student)
InstructorA	InstructorB
ClientInstructorA	ClientInstructorB

Access

First of all, we investigated who should have access to what. Students should, of course, have the authority to add and remove material in their own project. But they must under no circumstances be able to access any other project group’s information. The information from the instructors at Chalmers, for instance the course PM, bookmarks, templates and report synopses, should be readable for all students.

The instructors at Chalmers should have read access to all student projects, to be able to help them if necessary. They should also have the access to add material somewhere in the system.

The client instructors should have reading right to their supervised project as default, if necessary the access should be able to be extended, for instance if they need to share some information. For an overview of the access, see figure 5.1.

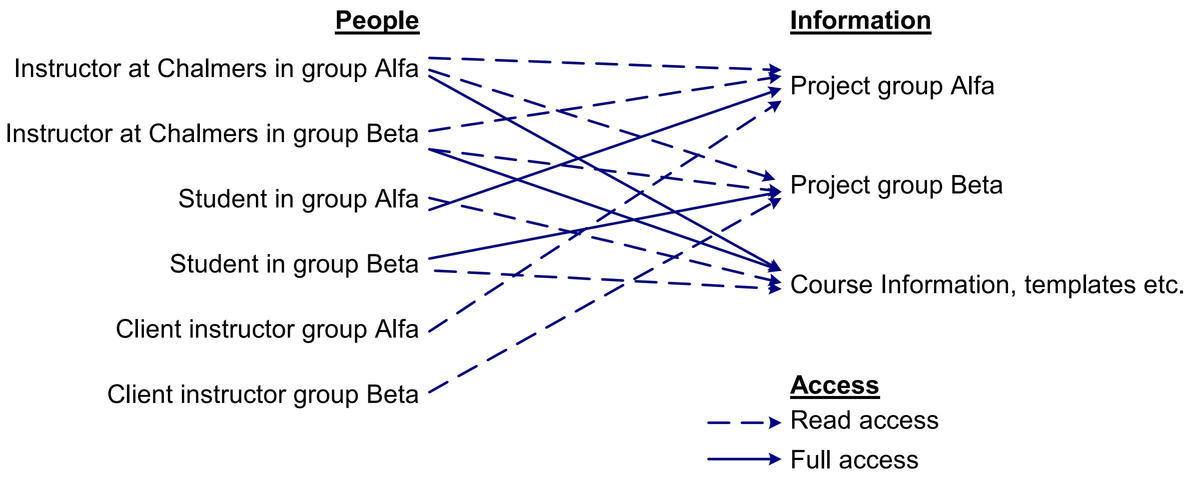


Figure 5.1: Schematic figure over the different roles access.

Fictional companies were created and connected to each of the groups, to get a clear boundary between them. The instructors were placed in a company of their own, called Chalmers, to be able to give them access to all student projects. The client instructors were placed under their project group company as well as the students.

The different role gives different authorities and permissions, the students and the instructors should not be able to do or access the same things. In eMatrix there are a large number of different roles available. An investigation of those resulted in that only three of them are needed: Project User, Project Lead, and Access Grantor. The Project User role enables the person to use the eMatrix Program Central and the Project Lead is the role that enables the person to change preferences in a project and to create projects. A person with the Access Grantor role has the privilege to access almost everything. All people have to have the Project User role to get access to Program Central. The students and the instructors at Chalmers should also have the Project Lead role to be able to edit preferences in their project. The client instructor on the other hand should only be able to have read access in the project. Finally the instructors at Chalmers also needs the Access Grantor role, this since they must have access to all projects to be able to instruct the project groups with their different special competence. Unfortunately, the Access Grantor role also gives access to add and remove information in the student projects, but this is not a major problem. The only alternative to give the instructors the Access Grantor is to add them in the project, which can give the students the feel of being too much supervised. For the different category roles, see table 5.2.

Table 5.2: The roles assigned to the different categories.

Students	Instructors	Client Instructors
Project User Project Lead	Project User Project Lead Access Grantor	Project User

Time plan

eMatrix Program Central has a functionality called WBS (Work Breakdown Structure), and it is an advanced time plan. In the WBS is it possible to create phases and task, and they have a specific lifecycle. The WBS gives, if used properly, a good overview of the project. The state together with the "% Completed" column shows in which phase and how much is done so far. The PD course is divided into three phases; pre study, concept study and detailed development phase. These phases contain different tasks depending of what should be done in the phases. In the system the phases are created, see figure 5.2. The students can then, by themselves, specify and break down the activities into tasks.

						Estimated		
Task Name	WBS	Task Type		State	% Complete	Duration	Start Date	End Date
Laptop Alarm		Project Space		Active	0.0	145.0	Oct 28, 2002	May 16, 2003
Pre Study	2	Phase		Assign	0.0	35.0	Oct 28, 2002	Dec 13, 2002
Concept Study	3	Phase		Assign	0.0	70.0	Dec 16, 2002	Mar 21, 2003
Detailed Development	4	Phase		Assign	0.0	40.0	Mar 24, 2003	May 16, 2003

Figure 5.2: The WBS structure before the student's changes.

The time plan should make it visible for the project members and other that may be concerned how the project should be carried out. The plan is also an instrument illustrating what time is needed to achieve the goals with the project and who is responsible for the specific tasks. The WBS in eMatrix Program Central is rather hard to grasp and a better way to overview the process is for instance a Gantt chart. To create a Gantt chart in the system is not possible, however it is possible to export the WBS to MS Excel. The data in MS Excel is presented in a table, from where a Gantt chart manually, with a small amount of effort, can be created.

Folder structure

To store data in a structured way is a core function in a PDM system. To be able to manage different versions is another key function. To structure the project properly, an investigation of the issues needed to be supported was made. On the basis of the investigation a generic folder structure was created. The generic model includes the common issues, see figure 5.3, as the project changes over time and the knowledge increases the structure can be updated.

The included folders are Commission, where the assignment, market analysis, problem definition, requirements and stakeholder analysis is included. The Concept folder includes the different solutions and an evaluation of which concept to develop further. In the meeting folder the meeting protocols are to be placed. The product folder should be a kind of product tree with assemblies and parts. In the Prototype folder

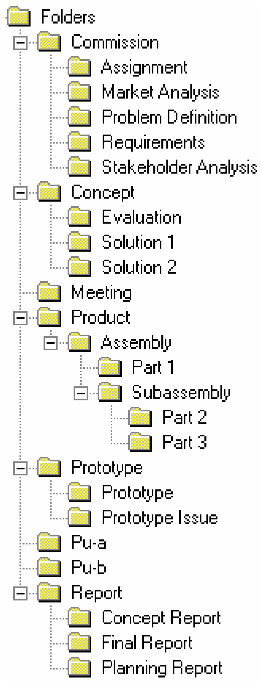


Figure 5.3: The generic folder structure

the prototype documents and issue should be placed. The PU-A and PU-B folders are created to enable storage of documents created in those courses. These courses are integrated in the PD course, and some of the mandatory assignments are directly coupled to the development project. Finally was the report folder created, where the mandatory reports, that should be done, will be placed.

Communication

The communication plays an important role in a project. In eMatrix Program Central there are three ways to communicate: by Iconmail, a simple mail function, by Discussions, a type of chat board, and by Routes, a process where tasks can be included. In the manual, appendix I, these three communication functionalities are described in detail.

Usability

To make the system as easy as possible to use, some changes has been made. The interface looks the same, but some menu alternatives have been removed, since they will not be used in the course, an example is the project tree menu, see figure 5.4.

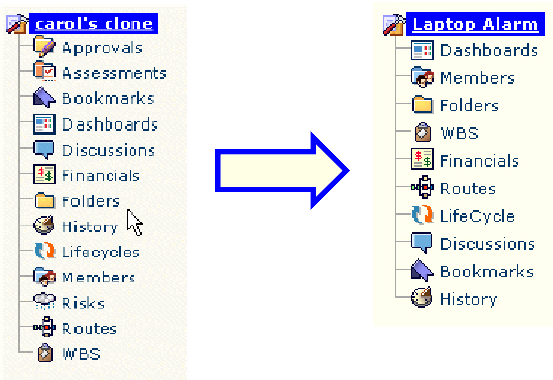


Figure 5.4: Changes made in the project tree menu.

The functionality that has been removed in the created environment is Approvals, Assessment and Risks. The approval functionality shows all approvals that have been made in the project level, for instance if a route governs whether the project can be promoted into next state or not. Since it is currently not possible to add content properly in the route functionality, the meaning with the approval functionality is not available, and has therefore been removed. The assessment functionality enables the project leader or a project member assigned as project assessor to evaluate the status of the project. The areas that can be assessed are: schedule, costs, resources and risks. The assessment functionality can be useful if it is used properly, but the functionality requires, in this kind of project, more resources than it gives support to the project group. The risk functionality enables the project leader to manage risks. Since this functionality also needs some time to learn it is removed. Another reason is that risk management is not given that much attention, and therefore is not highly prioritised by the students.

Requirements management

The intention was that some simple requirements management functionality should be developed in the system, but since the lack of time, due to problem to install the software, this task was excluded. The possibility to develop such functionality in the PDM system should, even though the time had not been limited, have been very demanding to fulfil.

This is because that kind of customisation would need a specific development tool, which was not available. This specific development tool, called Application Development Kit, is needed to enable major changes in the java applet code.

Concluding the PDM environment

The decision to only work in eMatrix Program Central, which is a module of the whole PDM system, lead to that some desired functionalities, which only exist in the other centrals, was disabled. These core functions are product structure management (PSM) and parts management, the functionalities are described in the frame of reference. The PSM functionality is desirable but the decision to exclude it was based on the complexity it would cause consequently leading to that the system could be to difficult to learn. The parts management on the other hand is not as important in the PD projects, since no reuse of design information is necessary or possible.

The model that has been implemented can be described as an advanced document manager and a project support tool. It supports the project groups to: structure the documents properly, develop and follow up the WBS, create and follow up the budget, share information and to work distributed.

The functionality that is removed from the system in the environment might be, if wanted, added in the future. The risk management is an interesting area, and if more weight is put into this area in the course in the future this functionality, with emphasis, ought to be added. The Assessment functionality might also be added in the future, this might be a tool for the instructors to give the project group a structured feedback of the status in the project. Before implementing any functionality it is important to be sure that the project group really benefits from using it, instead of just giving them an extra task to fulfil.

Project Templates

While working in project teams it is useful to find a standard way to handle project related information. To create project templates is one way to make the project work more standardized with a common frame of reference. When the project team has templates to lean to, is it possible for the group to have a stronger focus on the development instead of putting a lot of effort in project administration. This gives the project a fast start up process and it also decreases the risk of making mistakes that already been done. Another point of view is, why invent the wheel twice. Some work in the manner of the template thinking is based on “LIPS1 – nivå 1” [Svensson, Krysander, 2002].

PD course templates

The product development course has a flat organization, as was illustrated in the frame of reference chapter, and therefore is the need of templates not based on a standard procedure for reporting to, for instance the board of directors. Instead the template aims to speed up the development process and make it more standardized.

As was stated in the frame or references is the project development process divided into three main phases, these are: pre-study phase, concept study phase, and detailed development phase. The phases focus on different tasks and have therefore also different needs for templates.

Requirement specification

As was told in the frame of references the requirement specification should answer the question what the product has to do. A specification item consists of a metric and a value [Ulrich and Eppinger, 2000]. For example, “average time to assemble” is a metric, while “less than 75 seconds” is the value of the metric. Together, these form a specification and that is what a requirement specification is, a set of individual specifications from different stakeholders. The specification is based on the customer needs complemented with requirements from internal and external stakeholders, which is elicited with support from a product life-cycle template. An example of a requirements specification template is presented in appendix II.

Meeting protocol

Project meetings aim to clarify the project status, report on what has been done since the last meeting, what should be done until the next meeting, and by whom. Project meetings, in the PD course, can roughly be divided into two types, meetings with instructors and student group meetings, however they still have the same procedure. A template, that standardizes and simplifies this procedure, is a helpful support for the project leader. One example of an important issue in the template is the agenda, which should be put up to clarify the aim of the meeting. Other necessary issues are illustrated in the meeting protocol template, which is to be found in appendix III (in Swedish).

Project diary

In the project diary all things that have been done should briefly be written down. This should be done to get an overview of the activities performed, and to get the most out of the project. It is also good to have in the mandatory meeting with the teamwork specialist instructor (Psychologist).

A project diary is maybe not the first to think of when talking about project templates however is it useful to have and recently has the knowledge of its importance attracted attention. The diary is not made to blame anyone for mistakes. The project diary template is made as simple as possible, there are two rows meant as a calendar

Concept selection

Concept selection is the process of evaluating concepts with respect to customer needs and other criteria. It is suitable to split up the concept selection process into two steps with purpose to compare the relative strengths and weaknesses of the concepts. At first, all the concepts are evaluated according to the selected evaluating criteria and one or more concepts is selected for further investigation. The final element is to get more information about the selected concepts, which will function as a foundation for weighting and rating the concepts once more to choose the final concept. See figure 5.5.

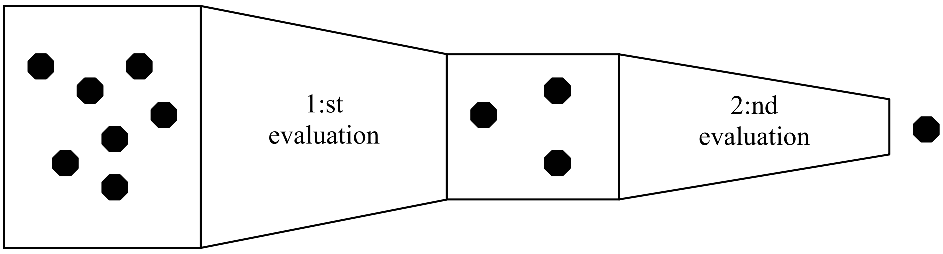


Figure 5.5: The concept selection process.

According to Ulrich and Eppinger [2000] all of the early phases in the product development process are extremely influential on eventual product success. A structured concept selection process helps to maintain objectivity throughout the concept process and guides the project group through critical and difficult processes. To support these two steps, evaluation matrices have been developed based upon Pugh [Ulrich and Eppinger, 2000]. The concept selection aims to support the first step, and the concept screening matrix aim to support the second. The complete templates are presented in appendix IV.

Failure Mode and Effect Analysis, FMEA

The FMEA method is used to find weaknesses in the developed product or in the manufacturing process, and to evaluate the effects if something goes wrong with a specific component or process. This process is difficult and can be very time consuming especially when the products are complex, therefore specialist tools are often used as support. In the PD course, were the products are not that complex, the need of supporting computer tools is maybe not that essential. However, the process is still important for the final result of the product and a template helps to structure the work. An example of a FMEA template can be found in appendix V.

Report synopsis

The report synopsis contains all the items that should be included in the final report. It is included to give the students an overview of the possible activities that can be done in the project. The report synopsis is put together by the PD course administration, and is available in appendix VI.

Viewer functionality

When designers and other members in the project team, at different companies as well as in the own organization, are collaborating in the development of a product they need to share and discuss geometrical information. To make it possible for all members to have an opinion and to avoid mistakes in the communication of the geometrical information it is necessary to give all involved the same information in the same way. This can be done either by giving all members access to the native software, which the information is created in, or by using a universal viewing tool. Giving all members access to all the different software can be very expensive and it is not necessary to give all users the full CAD functionality, when they only need an illustration tool. A better way, and far more cost efficient, is to use a universal viewing tool.

Surveys of collaborative design tools, which support viewing functionality in eMatrix, have been made in the thesis work “Collaborative Product Design Environments”

[Johansson and Uvhagen, 2001]. However all these software included in that survey have the drawback that they cost, therefore another tool has been added to the survey and investigated. The tool is a freeware from PTC for Pro/E CAD files, which is the CAD program used in the mechanical engineering education at Chalmers. In the investigation, only the most interesting evaluation criteria, from our point of view, have been used.

The criteria for which the program is evaluated are; Interface, viewing, markup, printing, measuring, visibility, real-time interaction, and licensing.

ProductView Express

ProductView Express is a java based web solution from PTC to view Pro/Engineer files only and is a "lightweight" version of ProductView that has more features and supports more file formats. The fact that the program can only handle one CAD format is a drawback, but because this particular software is used in the educational program this will not be regarded.

Viewing – Information has to be created in Pro/Engineer, which is the only format supported by the program. Dynamic spin, pan, zoom, and fly-through capabilities are actions that can be used to get a good view of the model. It is also possible to present the object in wireframe, hidden line, no hidden, and shaded state. The viewing functionality satisfies the basic needs that one could expect.

User interface – The ProductView Express is simple to use and easy to learn. The possibility to customize does not exist, though the size of the viewing area is adjustable. Se figure 5.6.

Figure 5.6: Sample of ProductView Express interface.

Markup – There is no possibility to make comments and to markup. This is a helpful functionality when communicating the opinions to the designer and vice versa. The absence of this functionality is a quite large disadvantage.

Printing – Printing is only supported through the web browser print functionality. This functionality may be wanted by some people, for example to bring a description to a meeting, but most people look at the model on the screen and gets their opinion there.

Measuring – This is important when engineers communicate with each other, for example when it is necessary to get exact dimensions from a model. In the ProductView Express application however it is not possible to make measures in the model.

Visibility – To make things clearer it is sometimes wanted to be able to cut out unnecessary details. In this application is it possible to "click" at specific parts in an assembly, to get a view of just that single component.

Real-time interaction – To have real time interaction often make the communication easier. The same model view is presented on the screen no matter where the project member is located. This functionality is very important when working in distributed teams. However, this functionality does not exist in ProductView Express.

Licensing – Unlike the other software in the survey, the licensing to ProductView Express is free.

Concluding the viewer functionality

To judge of the result from the survey the product is well suited for the type of projects carried out in the PD course. The viewer tool has limeted functionality or you could say that the only functionality it has is the possibility to view drawings and models. But that is also the only functionality that is necessary for the project members when they are working from home or a place where no native software exist.

A great benefit is that it is very easy to use and learn the program, it is self-instructive, which mean that the team members do not have to spend a lot of time on learning a new software. This is very good because the project members already have a complex program they have to get to know and do not need another software to learn. The focus in the course is on the product development and not on supportive computer tools, this has to be kept in mind.

There are some drawbacks with the program but they are all related to the lack of functionality. One of the most severe problems is when the client company is communicating files, which are not in PRO/Engineer format, with the project group. The group will then not be able to have viewer functionality for the models and drawings created by the client company, unless the first, if possible, convert them into Pro/E files.

In comparison with the other available software ProductViewer Express has a limited functionality. But the fact that the program is free and easy to learn makes it to our choice of viewer tool.

CHAPTER 6. TEST CASE

In the following chapter a test project is set up and the system is evaluated and discussed from a project member point of view.

Simulated project test

To evaluate if the developed environment was good enough for the given purpose a simulated assignment was given. The assignment was in short to come up with new door opening solution were no large force is needed when opening the door. The need had arrived because many old people feel that applying any significant force can be difficult and painful, especially for disabled people. The purpose with the study was to find weaknesses in the PDM system and the structure of information storage (folders) that has been built. To ensure that the system supports most of the product development lifecycle and no significant parts are missing the study has followed the stage gate process used in the PD course. As was described in the frame of references the stage gate process includes the following phases: pre study phase, concept study phase, and detailed development phase. The PDM system is evaluated on the basis of these classifications with an addition of the functionality supporting the whole development process.

In the pre study phase the most of the work is focused on understanding the assignment from the client company. Thus, what shall be done, what problem do we need to focus on, who is our client, what does the client want from us, and who is the customer of the product, are some questions that gave to be answered. In the early stages in the development process where the uncertainty is great, and the group have rather little help from the PDM system. The PDM system gives the project members the functionality of storage of information at a place available from any Internet connected computer. The result from this phase is a requirement specification set up on the basis of the stakeholder analysis. In the template folder, in the “clients at Chalmers project”, there is a template with a method that supports this important step to assure that nothing is forgotten.

The concept study phase is very much about generating ideas on the basis of the requirements specification, to brainstorm and come up with as many ideas as possible. In this stage it is important to take care of all ideas, their descriptions, and their underlying rationale. This work is also much about working together in the team and not about supportive computer tools. However the storage of all the concept ideas and the evaluations to get the final concept to go forth with is more of a PDM business. The loss of information decreases when having all information in one place. It would not be fun to loose the concept that could have been developed into the golden egg because of not having the documentation in order.

The detailed development phase is very much supported in the same way as the earlier stages. The support is about structuring the information and to tie the information where it belongs in the product breakdown. The file format of the files stored in the folders can vary, i.e. all file formats can be stored. But if the information should be useful we should be able to see it and then often the native software is required. The problem exists most of all when storing cad files and trying to see from a home computer where no native software is installed. In this case, this will not be a problem thanks to the product viewer tool that can handle this information and made it possible to work from home.

In the whole product development process the group is supported by the WBS, the communication tools, the folder structure helping to structure the information, and the financial support keep the budget in order. In the WBS is it possible to plan the project and to break down the phases into tasks. It is also possible to assign a task to a project member, who then is responsible for that it will be carried out correctly. This functionality makes the work more structured and also increases the responsibility in the group. A great drawback is the lack of possibilities to transform the information to a format easier to grasp. Communication functionality exists of three main types; routes, icon mail, and discussions but do not work satisfactory. When sending a route it is not possible, even though the option exists, to add content to it. The icon mail function for message deliveries and if an attachment will be added this is not possible. When working with folders it is hard to understand because the lack of information in the navigation tree. The financial functionality works very well and it is possible to get control over the budget.

Concluding the test case

A great drawback with the WBS is the lack of functionality transforming the information into a form that is easier to understand. However it is possible to export the WBS to MS Excel but still it will be some work to get it easier to view, like a Gantt chart for instance.

The folders are built up with subfolders and when a new folder is added it is just put under the category folder or subfolder. Thus, every time a folder or subfolder is added the structure grows with two levels. Another thing is that it is impossible to see if there exists a lower level without clicking the subfolder. Just adding a plus symbol when a folder has subfolders could solve this, as in MS Explorer for instance, see figure 6.1.

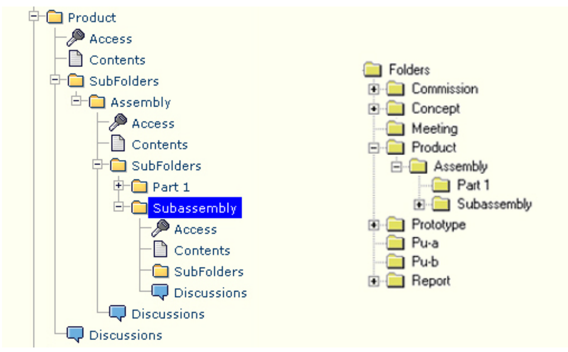


Figure 6.1: The difference between the folder structure in eMatrix and MS Explorer.

The route functionality is a powerful functionality if it just worked, but it does not. Sending a message a predetermined way functions, but when content is added to the route it is impossible for the receiver to open the added file and for example investigate it.

To work in small projects, like in the PD course, with relatively few project members it can feel like an overkill to have a PDM system defining how every little thing should be solved. Some times it can be a too structured way to work and at that time it is hard to see the benefits with implementing a system. And perhaps the drawbacks are more than the benefits but still the course is a preparation for how to work in development teams out in large companies using PDM systems in their daily work.

CHAPTER 7. CONCLUSIONS

In the introduction it was stated that Chalmers wants to get a step closer to the CDIO educational model. To do that the students more frequently have to combine theoretical and practical work, thus they have to learn the whole process to conceive, design, implement, and operate. To reach this goal the groups have to put a lot of effort and focus on solving the assignment given them and not fall behind the time schedule due to a lot of administrative work. By implementing a PDM system the team will be provided with a structured way to handle all project related information and a part of the CDIO goals can be reached.

Another reason to implement a PDM system in the PD course is to educate the student in how to use modern computer tools in the development process. 20 years ago it was CAD that supported the designers in the product development process, now it is PDM helping all the people in the extended enterprise to structure product and project related data.

An interesting issue that has arisen is, what is actually PDM? The meaning of the abbreviation is product data management and the purpose with a PDM system is to treat all product data during the products lifecycle, a more detailed description is available in the frame of reference. In this thesis, PDM stands more for project data management, that is to say that the system is more of a team support tool helping the project groups to structure the information and also to establish WBS, financials, and other helpful control tools guiding the projects in the right direction.

Have the objectives that were indicated in the introduction been achieved? The system should be up and running at the course start. That is not the case, however the first three weeks are the groups not helped by a PDM system and therefore is this point considered to be fulfilled.

Another issue is the image service that admits view functionality for CAD files. This task was solved with help from a recently published thesis work that had investigated several different 3D viewer software. A similar product was found and added to the survey. The product had very limited functionality but was evaluated regarding the most important criteria and found to be the best choice. The ProductView Express provides all the functionality that is required and most of all, it fulfil the most decisive criteria, it is free. The PDM system is accessible from any Internet connected computer and budget functionality exists. In other words, the system has the functionality that was required. This has been achieved very much due to the fact that the PDM system eMatrix is an out-of-the-box solution.

Only in one point of the objectives has the thesis work failed. This objective evolved due to the lack of requirements management functionality in existing PDM systems. There were two reasons for not being successful in this point. The first was the lack of time due to installation problems not under our control, and the second reason was because special development tools were required to develop this functionality in the web application, which was not available. However, even if the system had been installed in time it would still have been a very hard task to accomplish.

One important task was to make the environment easy to learn minimising the risk of scaring the students away with the image of the system as just another software to learn. As it is now, when some functionality has been removed, the environment has become less complex and better fitted to the type of project, as in the PD course. The functionality that has been removed from the system in the environment might be, if wanted, added in the future, but it is then important to be sure that the project group really benefits from using it, instead of giving them and extra task to fulfil.

To assure that there will be no trouble and no extra load for the project group a manual was made to guide the users when having problems. The overall judgement of the environment, without the possibility to consider the evaluation at the course end, is that it will be very successful.

CHAPTER 8. RECOMMENDATIONS FOR FUTURE WORK

At the end of the PD course the system and the use of the system has to be evaluated to see if the project groups where supported and helped in the team-based work. An interesting point of view is to connect the project result to the use of the system and see if groups that have used the system diligently have a larger amount of success in their projects. It is also of interest when the system has been used, if it is used continuously or just in short periods of time before for example a presentation.

To continue with the objective where this thesis work failed is also an interesting work for the future. To add the requirement management functionality would give greater possibilities to control the development and to make the right decisions in critical situations.

It would also be of great interest to investigate and implement Engineering Central in the PD course. The Engineering Central is more of a product management tool than a project management tool, as Program Central. A change like this would support the product development and make it possible to build up product structures with for example alternate and substitute parts. It would also be possible to create bill of materials (BOM) etc. see chapter 4 for a detailed description.

REFERENCES

Literature

Andersson, F., Nilsson, P. and Johannesson, H.L. (2000)
Computer based requirement and concept modelling – information gathering and classification, Proceedings of DETC'2000 ASME Design Engineering Technical Conference, Baltimore, MD, USA.

Andersson, F. (2001)
Functional Representation for Design Support A Design Theory Perspective, Licentiate thesis, Department of Product and Production Development, Chalmers University of Technology, Göteborg, Sweden.

CIMdata (2001)
collaborative Product Definition management (cPDm): An Overview, CIMdata inc., Ann Arbor, MI, USA.

CIMdata (1998)
Product Data Management: The Definition, CIMdata inc., Ann Arbor, MI, USA.

Davis, A., Hickey, A. (2002)
Do We Practice What We Preach?, Department of Information Systems, University of Colorado at Colorado Springs, Colorado Springs, Colorado, USA

Johannesson, H.L. (1997)
Application of Interaction and Functional Couplings Rules in Configuration Design, Proceedings of DETC'97 ASME Design Engineering Technical Conference, Sacramento, CA, USA.

Kotonya, G. and Sommerville, I. (1998)
Requirements Engineering, John Wiley & Sons Ltd, Chichester, UK.

Krysander, C., Svensson, T. (2002)
LIPS – nivå 1, Bokakademin, Linköping, Sweden.

Malmqvist, J. (2001)
Implementation of Requirements Management - A Task for PDM Systems or Specialized Software Tools? Systems Engineering: vol 4, no 1, pp 49-57.

Mendel, A. (1999)
PDM and the Internet, www.CIMdata.com

Mesihovic, S., Malmqvist, J., Pikosz, P. (2002)
Product Data Management (PDM) System Based Support For Engineering Project Management. Proceedings of TCME-2002, Wuhan, China.

Nilsson, P. (2001)
Towards an Integrated Product Model Supporting the Early Phases of Product Development. Lic.Eng. Thesis, Product and Production Development, Chalmers University of Technology, Göteborg, Sweden

Philpotts, M. (1998)
The Benefits of Product Data Management, www.cimdata.com

Schachinger, P., Johannesson, HL. (2000)
Computer Modeling of Design Specifications, Journal of Engineering Design, 2000 vol. 11, no. 4, p. 317-329

Sommerville, I. and Sawyer, P. (1997)
Requirements engineering A good practice guide, John Wiley & Sons Ltd, Chichester, England.

Stark, J. (2000)
Product Data Management (PDM), www.johnstark.com.

Stevens et al. (1997)
System engineering: coping with complexity, Prentice-Hall Europe, London, UK.

Sutinen, K. (2001)
Requirements Driven Product and Process Model – Supporting Concept Development, Licentiate thesis, Department of Product and Production Development, Chalmers University of Technology, Göteborg, Sweden.

Svensén, U. et al (2000)
Global Study on Product Development, ENDREA, Sveriges Tekniska Attachéer, Stockholm, Sweden.

Ulrich, K., Eppinger, S. (2000)
Product Design And Development, 2nd ed., McGraw-Hill Companies, Inc., USA.

Internet

Nationalencyklopedin på Internet, www.ne.se, 2002-08-21

CDIO, www.cdio.org, 2002-06-01

Cimdata, www.cimdata.com, 2002-12-09

Incose, www.incose.org, 2002-12-09.

Technia, www.technia.com, 2002-09-04

Appendix I

Användarmanual

för eMatrix

till kursen Produktutveckling (MMF 205)

Appendix I

Innehållsförteckning

Appendix I

Introduktion

För att förbättra förutsättningarna i projektarbetet har kursansvariga i den här kursen beslutat att ge projektgrupperna tillgång till PDM-systemet eMatrix. Detta ger projektgruppen bättre möjligheter att dela produktdata, hålla reda på filer och strukturera sitt projekt.

Den här versionen av eMatrix är webbaserad, till skillnad från den version som några av er arbetat med i CAD&PDM kursen, vilket betyder att man kan koppla upp sig var som helst bara man har tillgång till internet och att den är mycket mer användarvänlig. Kraven på datorn är inte stora, däremot måste minst Internet Explorer 6 eller Netscape 4.7 vara installerat.

Logga in

1. Öppna webbrowsern och skriv in:
   http://ny07.mtek.chalmers.se:8001/ematrix/emxLogin.jsp

2. Skriv in användarnamn och lösenord.
   Fälten är case sensitive, så var noggrann med stora och små bokstäver.

3. Klicka **Login**.

Inloggningsinformation finns på sida 6.

Appendix I

Gränssnittet

När man loggat in ser sidan ut som nedan. Uppe till höger finns **verktygsfältet**, till vänster finns **applikationsmenyn** och det beiga fältet är **arbetsytan**. Man kan även se vem man är inloggad som.

Verktygsfältet

Verktygsfältet finns tillgänglig hela tiden, den innehåller viktiga funktioner som mail, ändra lösenord och logout m.fl.

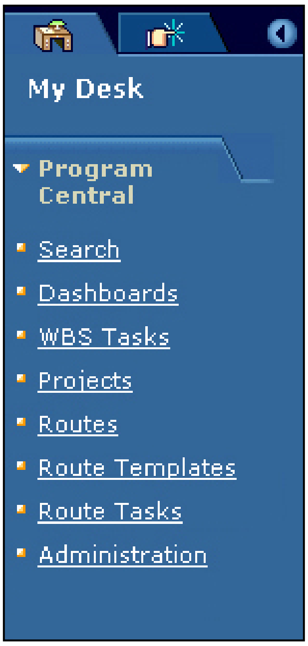
	**Home** – Tar dig till din startsida		**Change password** – Ändring av lösenord
	**Preferences** – Val av startsida		**Application menu** – Ger tillgång till gamla centraler (används ej i kursen)
	**Iconmail** – Intern mailhanterare		**Logout** – Loggar ur och tar dig till inloggningssidan

Applikationsmenyn

I applikationsmenyn finns de flesta funktionerna som finns i eMatrix. Menyn är uppbyggd av två flikar, ”My Desk” och ”Actions”. Beroende på vem man är inloggad som finns olika antal centraler i menyn. Men som projektmedlem används endast ”Program Central”. I Program Central jobbar man med projekt i grupper, såväl inom företaget som tillsammans med andra företag.

Appendix I

My Desk



Search – Sökning av personer, projekt, dokument mm

Dashboards –Visar en översikt över flera projekt

WBS Tasks – Visar uppgifter man tilldelats i projektet

Projects – Visar vilka projekt man ingår i

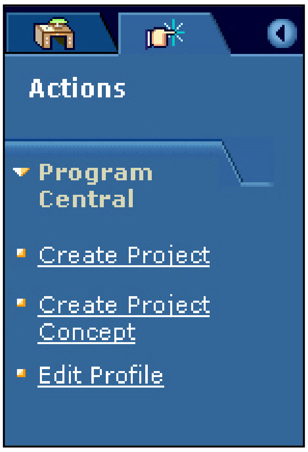
Routes – Visar rutterna man skapat och dess status

Route Templates – Visar tillgängliga mallar för att skapa rutter

Route Tasks – Visar vilka uppgifter man har fått från rutter

Administration – Ändra företagsdata, lägga till personer.

Actions



Create Project – Skapa projekt

Create Project Concept – Skapa pre projekt

Edit profile – Ändra inställningar i sin profil.

Appendix I

Ändra personliga inställningar

Första gången man loggar in med sitt tilldelade inloggningsnamn är det viktigt att byta password och fylla i kompletterande personuppgifter. Detta för att alla från början har samma password (pdmn). Inloggningsnamnet är det samma som början på e-mailen (**pdmn**@mtek.chalmers.se blir således **pdmn**). Det finns dock några undantag, se bifogat papper på handledningstillfället.

Byte av password:

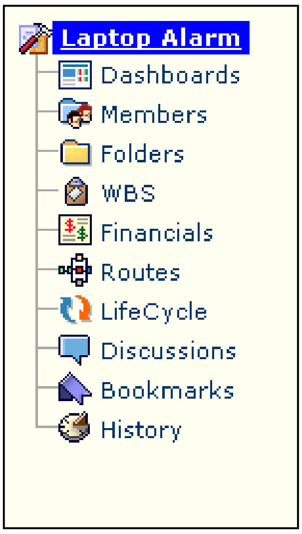
1. Klicka på nyckelsymbolen i verktygs fältet
2. Fyll i nuvarande password (pdmn)
3. Välj nytt password, verifiera och klicka på **Done.**

Editera personuppgifter:

1. Klicka på Actionfliken i Applikationsmenyn
2. Klicka på **Edit Profile**
3. Fyll i dina personuppgifter och klicka på **Done**.

Jobba med projekt

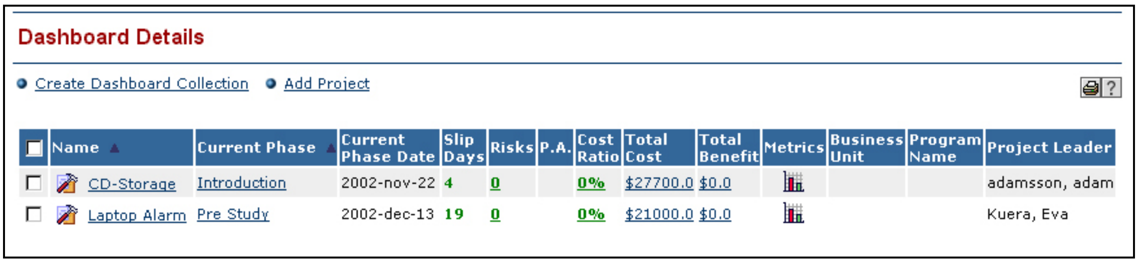
Projektet som gruppen skall jobba med är redan skapat. För att komma till det loggar man in, klickar på **Projects** i Applikationsmenyn och väljer projektet som gruppen tilldelats. När detta är gjort kommer projektträdet upp på arbetsytan.



Dashboards – Visar en översikt av projektet
Members – Editering av projektmedlemmar och deras roller
Folders – Foldrar och subfoldrar för lagring av projektmaterial
WBS – Work breakdown structure
Financials – Hantering av projektbudgeten
Routes – Visar rutter inom projektet
LifeCycle – Visar projektets livscykel och dess status
Discussions – Diskussionsforum för projektmedlemmarna
Bookmarks – Visar länkar till information som är intressant för projektet
History – Registrerar ändringar som gjorts i projektet.

Dashboards

Dashboarden visar en översikt av projektet. Information som man kan se på dashboarden är vilken fas projektet befinner sig i, om man ligger i fas eller efter, översikt av budgeten samt vem som är projektledare för tillfället.



Dashboard Details

Create Dashboard Collection · Add Project

	Name	Current Phase	Current Phase Date	Slip Days	Risks	P.A.	Cost Ratio	Total Cost	Total Benefit	Metrics	Business Unit	Program Name	Project Leader
	CD-Storage	Introduction	2002-nov-22	4	0		0%	$27700.0	$0.0				adamsson, adam
	Laptop Alarm	Pre Study	2002-dec-13	19	0		0%	$21000.0	$0.0				Kuera, Eva

Members

Här finns alla gruppmedlemmar i projektet. Det går även att se vilken behörighet olika personer har i projektet (access) samt vilken roll de olika personerna har (role). Man kan även lägga till personer till projektet.

Appendix I

Lägga till personer till projektet:

1. Klicka på Add

2. Välj sökdata och klicka på Find

3. Välj person att lägga till, avsluta med att klicka på Done.

Ändra access och roll

1. Klicka på Edit

2. Ändra det som önskas och klicka på Done.

Folders

En folder innehåller en fil eller flera filer som är relaterade och berör ett specifikt område. Foldrar kan också ha subfoldrar som i sin tur har subfoldrar, vilket gör det möjligt att bygga upp en mappstruktur för att organisera informationen. Denna struktur är redan uppbyggd i projekten, se figuren nedan, men allteftersom projektet framskrider kan ändringar behöva göras på grund av t ex förändringar i projektet eller ny kunskap som tillkommit. Foldrar och subfoldrar kan då behöva raderas eller läggas till. Exempel på hur det kan se ut finns i projektet CD-Storage. För att komma in i projektet så ska ni logga in som Username adam och lämna Password-fältet tomt.

Att lägga in dokument, skisser, ritningar och dylikt i foldrar fungerar ungefär som i utforskaren fast i eMatrix är det mer uppstyrt. Det beror på att man på så sätt får mer koll på sitt material. Man kan få reda på vem som gjort ändringar och om någon håller på att jobba med en fil.

Appendix I

Att skapa en folder:

1. Välj ditt projekt och folder i navigationsträdet

2. Välj **Create**, skriv i namn på foldern och beskrivning

Välj den typ av access du vill att foldern ska ha. Du kan tillåta att alla kan läsa den (Global Read Access) eller så kan du tillåta att endast projektmedlemmar behörighet att läsa (Project Space Read Access).

3. Klicka på **Finish**

Nu är foldern skapad med det återstår att lägga till access för projektmedlemmarna.

4. Klicka på **Access** och därefter på **Add Project Member(s)**

5. Bocka för de personer som ska få behörighet till foldern samt ge dem någon av de accesser som finns att tillgå

6. Klicka på **Done**.

Lägga till filer till en folder:

1. Välj **Folders** från navigationsträdet. Alla foldrar som du har läsrättigheter till listas upp. För varje folder visas namnet på foldern och hur många filer som har lagts till.

2. Välj den foldern som du vill lägga till en fil till.

3. Välj **Contents** från navigationsträdet.

4. Välj **Add Project Contents** eller **Upload External File**.

Add Project Content kan vara en användbar funktion i slutet av projektet när alla dokument ska sammanställas. Upload External File används, precis som namnet säger, vid uppladdning av externa filer till systemet.

5. Bläddra fram den fil du vill lägga till och klicka på **Done**.

Filen ligger nu i den folder du valt att placera den och klickar du på Content i navigeringsträdet så dyker följande bild upp.

		Title ▲	File Name	Type	Revision	Version	Date	Action	
		2002-11-15	2002-11-15.doc	Document	0	1	2002-nov-15		

Appendix I

Hur figuren ovan ser ut beror på vilken access man har till foldern. Om ägaren till foldern lagt till dig med fulla rättigheter så ser den ut enligt ovan. Om inte så finns endast möjligheten att ladda ner filen och titta på den men man har ingen möjlighet att ladda upp den i systemet igen. Detta syns i Action fältet då endast Download alternativet finns selekterbar.

 **Locked** – Om denna symbol syns så innebär det att någon har checkat ut filen och att det endast går att titta på den genom att klicka på Download. Denna säkerhetsåtgärd har man gjord för att inte två användare ska göra ändringar samtidigt utan vetskap om att någon annan håller på att göra ändringar i samma fil.

Title – Det namn du valt att filen ska kallas i systemet.

File name – Det namn den incheckade filen har och vad det är för filformat.

Type – I Program Central finns bara en type och det är Document. Dokumenten kan vara 3D cad filer, Excel filer eller av något annat format.

Revision – Skrivs som en kombination av revision och version t ex 0:1 vilket innebär att det är revision 0 och version 1. Revision är en bokstavssekvens med början på A, bortsett från originalfilen som har revision 0.

Version – Versionsnummer på filen ändras varje gång man checkar in en fil. Se Actions nedan.

Actions – Möjliggör bland annat in- och utcheckning av filer.

 Checkout – Om en fil är incheckad kan den checkas ut genom att man klickar på denna ikon. Det blir då möjligt att uppdatera filen.

 Checkin – Om en fil är utcheckad så klickar man på denna ikon när man åter vill checka in den i systemet. Ändringar som gjorts sparas i den nya versionen och 1 i versionsfältet ändras till en 2. Båda versionerna finns dock sparade i systemet och kan granskas när som helst. Detta gör man genom att klicka Title och Versions i det navigationsträd som dyker upp.

 Undo Checkout – Om man ångrar sig och ej vill göra några ändringar klickar man på denna ikon och allt återställs.

 Download – Om man vill ladda ner filen för att titta på den eller spara ner den på sin dator. I projektet PU 02-03 finns en del användbara mallar. Dessa hämtas hem genom att man går in i projektet klickar på foldrar, templates, content, väljer de mallar man vill ha och klickar på Download ikonen.

Appendix I

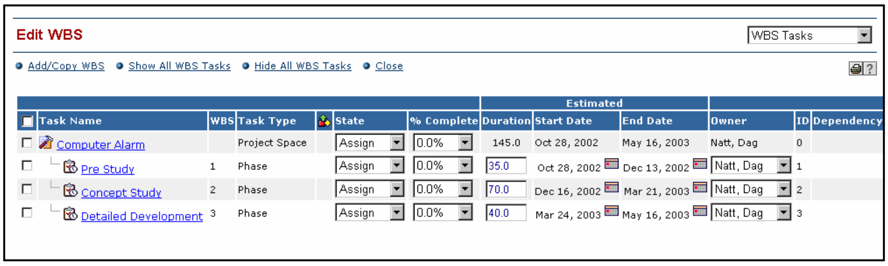
WBS – Work breakdown structure

I Work breakdown structure (tidsplanen) skapas projektstrukturen som kan liknas vid en hierarki av uppgifter som ska utföras under projektets gång. Genom att använda WBS i tvärfunktionella projekt ges möjligheterna att tilldela projektmedlemmar uppgifter samt att ge en översikt av hur projektet fortlöper. Ett vanligt misstag när en WBS görs är att man delar upp huvudmålen i för små och för många deluppgifter, vilket leder till att man istället skapar en ”to-do” lista som inte ger möjlighet till arbete med ansvar. En generell regel är att projektet bör brytas ner i uppgifter (tasks), där uppgifterna på lägsta nivå varar mellan en till åtta veckor.

Skapa och editera WBS

1. Klicka på **Projects** i användarmenyn.

2. Klicka på projektet du arbetar med.

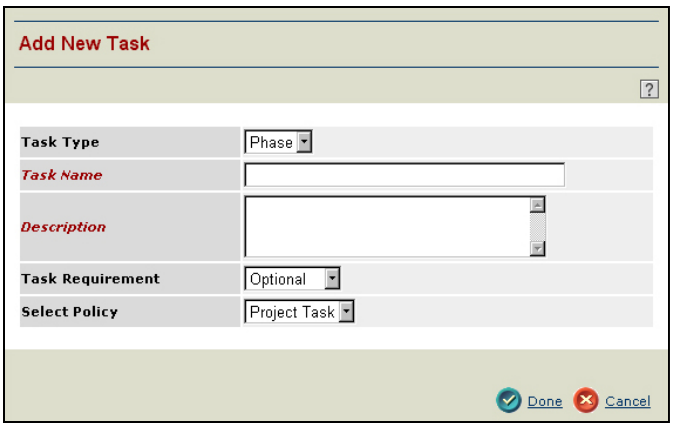


3. Klicka på **WBS** i trädstrukturen.

4. Klicka på **Edit** för att skapa/editera strukturen.

5. Bocka för projektet och klicka på **Insert At Selected**, vilket innebär att en Phase/Task läggs till i nivån <u>under</u> den förbockade.

Ett nytt fönster dyker upp så att en ny Phase/Task kan läggas till i strukturen.

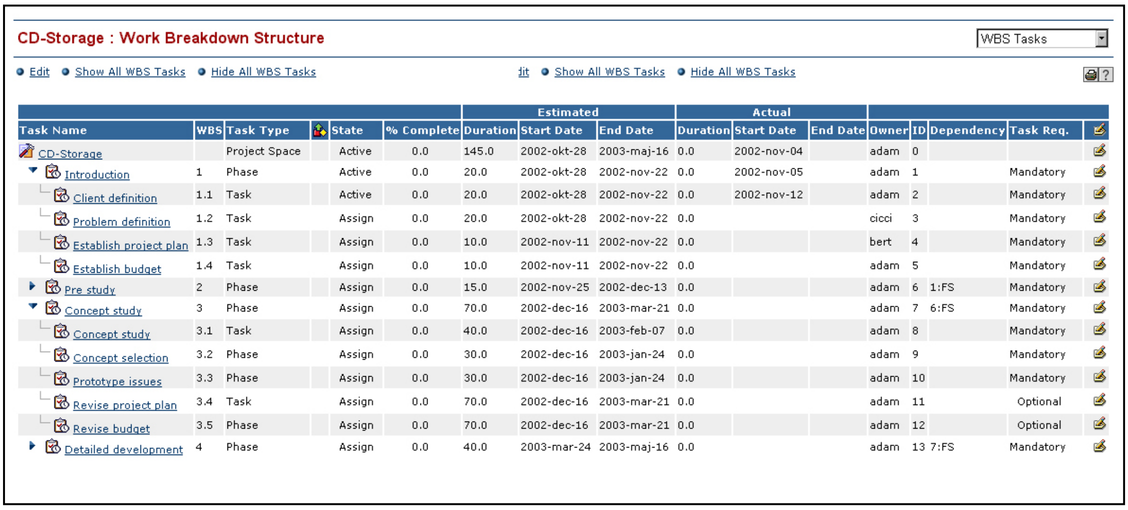


Appendix I

6. Välj **Task Type**, fyll i **Task Name** och **Description** samt välj om uppgiften ska vara **Mandatory** eller **Optional**.

7. Klicka på **Done**.

Skapa sedan alla Phases/Tasks som ska ingå i projektet, det görs enligt samma procedur som ovan. Efter det kan WBS-trädet se ut som nedan.



CD-Storage : Work Breakdown Structure

WBS Tasks

Edit · Show All WBS Tasks · Hide All WBS Tasks · dit · Show All WBS Tasks · Hide All WBS Tasks

						Estimated			Actual							
Task Name	WBS	Task Type		State	% Complete	Duration	Start Date	End Date	Duration	Start Date	End Date	Owner	ID	Dependency	Task Req.	
CD-Storage		Project Space		Active	0.0	145.0	2002-okt-28	2003-maj-16	0.0	2002-nov-04		adam	0			
Introduction	1	Phase		Active	0.0	20.0	2002-okt-28	2002-nov-22	0.0	2002-nov-05		adam	1		Mandatory	
Client definition	1.1	Task		Active	0.0	20.0	2002-okt-28	2002-nov-22	0.0	2002-nov-12		adam	2		Mandatory	
Problem definition	1.2	Task		Assign	0.0	20.0	2002-okt-28	2002-nov-22	0.0			cicci	3		Mandatory	
Establish project plan	1.3	Task		Assign	0.0	10.0	2002-nov-11	2002-nov-22	0.0			bert	4		Mandatory	
Establish budget	1.4	Task		Assign	0.0	10.0	2002-nov-11	2002-nov-22	0.0			adam	5		Mandatory	
Pre study	2	Phase		Assign	0.0	15.0	2002-nov-25	2002-dec-13	0.0			adam	6	1:FS	Mandatory	
Concept study	3	Phase		Assign	0.0	70.0	2002-dec-16	2003-mar-21	0.0			adam	7	6:FS	Mandatory	
Concept study	3.1	Task		Assign	0.0	40.0	2002-dec-16	2003-feb-07	0.0			adam	8		Mandatory	
Concept selection	3.2	Phase		Assign	0.0	30.0	2002-dec-16	2003-jan-24	0.0			adam	9		Mandatory	
Prototype issues	3.3	Phase		Assign	0.0	30.0	2002-dec-16	2003-jan-24	0.0			adam	10		Mandatory	
Revise project plan	3.4	Task		Assign	0.0	70.0	2002-dec-16	2003-mar-21	0.0			adam	11		Optional	
Revise budget	3.5	Phase		Assign	0.0	70.0	2002-dec-16	2003-mar-21	0.0			adam	12		Optional	
Detailed development	4	Phase		Assign	0.0	40.0	2003-mar-24	2003-maj-16	0.0			adam	13	7:FS	Mandatory	

Appendix I

Ändra start- och slutdatum

Nu när trädet är skapat ska man tidsbestämma uppgifterna.

1. Välj **Start Date** genom att klicka på kalender-knappen och sedan välja datum.

2. Välj **Duration**, alltså hur länge uppgiften ska hålla på, och klicka med musen någonstans på skärmen.

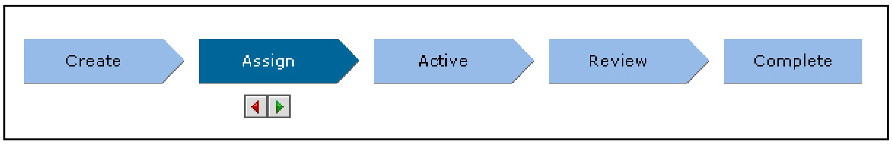
Tyvärr kan man inte välja start- och slutdatum utan man måsta välja startdatum och sedan, för hand eller prova sig fram, välja antal dagar uppgiften får ta för att det ska bli riktigt. Endast uppgifterna i den understa nivån kan tidbestämmas, eftersom de andra byggs upp av dem. Ett exempel på en WBS finns skapad i projektet CD storage. För att komma till detta projekt, logga in som adam, inget password.

Ändra State

I projektets WBS sätter man upp planerade start- och slutdatum för uppgifterna, men det mest intressanta är ändå de faktiska datumen.

Actual Start Date är den datum då uppgiftens state i livscykeln promotas till **Active**.

Actual Finish Date är den datumen då uppgiften promotas till **Complete**.



Man kan ändra uppgiftens state (tillstånd) på två sätt. Antingen genom att klicka på edit knap och välja state, eller genom att klicka på lifecycle i WBS-trädet och där promota till nästa state.

Ändra % Completed

När man arbetar med en uppgift kommer man, förhoppningsvis, närmare och närmare slutet. För att visa hur man ligger till, ändrar man hur stor del av uppgiften som är klar.

1. Klicka på **WBS** i projektträdet.

2. Klicka på edit knappen längst till höger i raden för uppgiften.

3. Ändra **% Completed** till det nya värdet.

4. Klicka på **Done**.

Varje task/phase som skapas i WBS kan man, genom att klicka på den, jobba med som ett delprojekt. När man klickar på den öppnas ett nytt fönster, där ett Review Tree visas.

I WBS-trädet finns ungefär samma saker som finns i projektträdet, det som är nytt här är **Deliverables** och **Dependencies**. I **Deliverables** lägger man saker som skall "levereras" eller vara klara när uppgiften är klar. Man kan antingen lägga in nytt material eller, vilket föredras, länka existerande material hit. I **Dependencies** kan man skapa beroendeskap, till exempel att en uppgift måste vara klar innan en annan kan starta. Det finns fyra olika beroendeskap:

- FS – (finish to start) Uppgiften kan inte startas innan den föregående avslutats
- SF – (start to finish) Uppgiften kan inte avslutas innan den föregående startats
- FF – (finish to finish) Föregående uppgift kan inte avslutas innan denna startats
- SS – (start to start) Uppgiften kan inte startas innan den föregående startats.

Exportera WBS

När WBS'en är klar kan man exportera den till Excel, detta för att till exempel manuellt skapa ett mer lättöverskådligt Gant schema. WBS'en är dock fortfarande nödvändig för att strukturera, fördela och följa upp projektets faser och uppgifter.

1. Klicka på **Projects** i applicationsmenyn.

2. Bocka för projektet där WBS'en finns och klicka på Export.

3. Fyll i filnamn och behåll övriga inställningar. Klicka på **Next**.

Excel öppnas och WBS'en uppenbarar sig i en enda stor "klump". För att det sksa bli överskådligt måste vissa ändringar göras.

4. Markera allt och gå in i menyn **Data**, välj **Text to Columns** och följ instruktionerna.

Financials

I eMatrix kan man även lägga upp en budget där man specificerar planerade utgifter och vinster, och efter en månad avslutats även föra in de verkliga siffrorna.

Appendix I

Skapa budget (planned)

1. Klicka på **Financials** i projekt-trädet

2. Klicka på **Create New**

3. Välj start tid och varaktighet för kostnaderna respektive vinsterna och klicka på **Next**

4. Välj vilka kostnadskategorier som är intressanta, klicka på **Next**

5. Välj vilka vinstkategorier som är intressanta, klicka på **Done**

6. Klicka på **Cost Categories** och klicka sedan på den kategori du vill planera. Välj **Edit**.

7. Fyll i de planerade kostnaderna (planned) och klicka på **Done**.

När alla kategorier är planerade återstår det endast att frysa budgeten, detta behöver göras för att budgeten ska kunna revideras och den verkliga kostnaden ska kunna fyllas i.

8. Klicka på **Lifecycle** i Budget trädet.

9. Promota till **Plan Frozen**.

Revidera budget (estimated)

Efter ett tag är det dax att revidera budgeten (estimated cost)

1. Klicka på **Financials**, **Project Budget**, **Cost categories.** Välj kategorierna i tur och ordning.

2. Fyll i de reviderade kostnaderna i **Estimated Cost** och klicka på **Done.**

När månaden påbörjats och den uppskattade budgeten fyllt i, fryses värdet och kan inte längre justeras.

Verklig budget (actual)

När månaden är slut ska den faktiska kostnaden (actual cost) fyllas i.

3. Klicka på **Financials**, **Project Budget**, **Cost categories.** Välj kategorierna i tur och ordning

4. Fyll i de faktiska kostnaderna i **Actual Cost** och klicka på **Done.**

När månaden avslutats och den verkliga budgeten fyllt i, fryses värdet och kan inte längre justeras.

Man kan även lägga till dokument i budgeten, det gör man genom att klicka på **Attachments**. Proceduren är densamma som att lägga till filer i folder.

LifeCycle

Livscykeln visar vilket State som projektet befinner sig i. När man skapar projektet befinner det sig i Create. När man håller på att sätta upp och delegera projektet ska det vara i **Assign** läge. När man startar projektet promotar man det till **Active** läge. När man är klar med projektet skickar man det till **Review** läge. Sen utvärderas projektet om det anses som färdigt promotas det till **Complete** läge. När projektet ska läggas på ”hyllan” skickas det till **Archive**.

Discussions

Diskussionsforum är precis som ett klotterplank. Man kan skapa nya trådar eller svara på existerande.

För att skapa ny tråd klickar man på **Post Message** och för att svara på en tråd går man in på meddelandet och klickar på **Reply**

Bookmarks

I Bookmarks kan man lägga in länkar till bra sidor på nätet. Det funkar precis som favoriter i Internet Explorer.

Lägga in en Bookmark

1. Klicka på **Bookmarks** och sedan på **Create New**.

2. Fyll i **Bookmark Name**, **URL** och beskrivning. Klicka sen på **Done**.

Kommunikation

PDM system bygger generellt på att man kommunicerar produktdata eller annan information som är knuten till produktens livscykel. Informationen är synlig för olika personer, beroende på hur relevant det är för personen och dennes arbetsuppgifter, inom företaget eller som t ex en underleverantör. Det finns även behov av att kommunicera genom att föra en dialog med andra parter. I eMatrix kan kommunikation ske antingen med Iconmail, Discussions eller Routes.

Icon mail

Iconmail fungerar på samma sätt som vanliga mail-program men med betydligt mindre funktionalitet, vilket innebär att man endast kan skicka meddelanden till personer som återfinns i företaget (Projektet) eller i det företag man samarbetar med (Chalmers).

Iconmailen är inte enbart för kommunikation genom en dialog utan fungerar även som en kommunikationskanal från eMatrix. Då du t ex har blivit tilldelad en uppgift (task) skickar eMatrix en ”notification” till dig med en länk till information om vad du förväntas göra.

Discussions

Discussions är ett diskussionsforum som är kopplat till de flesta objekt som skapas i eMatrix och återfinns bland annat i projektträdet. Diskussionen, som är kopplad till objektet, förs enbart på samma nivå som objektet. Projektgruppen bör därför ha en strategi för hur man ska använda forumet. Man bör t ex meddela de andra gruppmedlemmarna var diskussionen har startats.

Routes

Routes är det mest komplexa dialogverktyget av dessa och ger projektgruppen möjlighet att skapa egna kommunikationsprocesser och vägar (rutter). En rutt (route) är en grupp av uppgifter som projektmedlemmar ska utföra vid genomförandet av en aktivitet. Här finns också möjlighet att skapa mallar, för processer som upprepas, vilket effektiviserar arbetet.

Skapa routes

1. Klicka på **Routes** i projektträdet eller i applikationsmenyn och sedan på **Create New Route**

2. Fyll i ruttnamnet, beskrivning och ”Route Action”. Klicka på **Done**

Ett Add Content fönster dyker upp på skärmen, vilket ger möjlighet att bifoga objekt.

3. Add Content: Om något objekt ska bifogas klicka på **Add Content** och leta upp objektet. Om inget objekt ska bifogas klicka på **Done**.

4. Klicka på **Add Members** för att välja vilka som ska ingå i rutten. Klicka på **Done**.

5. Assign Task: Välj ordning (**order**) för hur uppgifterna skall distribueras.
   Fyll i:

Appendix I

Task Name – namn på uppgiften.

Name Action – välj namn på den person som ska tilldelas uppgiften samt vilken åtgärd denne ska göra. Det finns fyra åtgärdstyper:

Comment – mottagaren kommenterar innehållet,
Approve – mottagaren måste godkänna eller avvisa,
Notify Only – upplyser endast mottagaren och
Investigate – som comment fast mer utredande karaktär.

Instruction – fyll i instruktioner för rutten.

Due Date – fyll i förfallodatum för uppgiften och klicka på **Next**.

6. En översikt av vilka som är inkluderade i rutten, vilken ordning och vilka åtgärder de ska vidta presenteras. Klicka på **Next** om allt är i sin ordning.

7. Sista steget i skapandet av rutten väljs om rutten ska startas med en gång (Start Route) eller om rutten ska startas manuellt (Start Route Manually). Avsluta med att klicka **Done**.

Rutten är nu skapad och de personer som är inkluderade i den får ett icon mail och en task där de instrueras vad som avsändaren vill de ska göra. Om man vill editera, spara rutten som mall eller starta den manuellt klickar man på rutten, i route listan, och väljer det alternativ som önskas.

Appendix II

Chalmers	Kravspecifikation:					
Utfärdare:	Skapad:	Modifierad:	K/Ö	Vikt	Intressenter
Funktion:					
1. Prestanda					
1.1					
1.2					
1.3					
2. Miljö					
2.1					
2.2					
2.3					
3. Livslängd					
3.1					
3.2					
3.3					
4. Underhåll					
4.1					
4.2					
4.3					
5. Tillverkningskostnad					
5.1					
5.2					
5.3					
6. Konkurrerande lösningar					
6.1					
6.2					
6.3					
7. Distributionssätt					
7.1					
7.2					
7.3					
8. Paketering					
8.1					
8.2					
8.3					
9. Kvantitet					
9.1					
9.2					
9.3					

Appendix II

10. Tillverkningsanläggning			
10.1			
10.2			
10.3			
11. Storlek			
11.1			
11.2			
11.3			
12. Vikt			
12.1			
12.2			
12.3			
13. Estetik och ytfinish			
13.1			
13.2			
13.3			
14. Material			
14.1			
14.2			
14.3			
15. Produktlivscykel			
15.1			
15.2			
15.3			
16. Standarder och lagkav			
16.1			
16.2			
16.3			
17. Ergonomi			
17.1			
17.2			
17.3			
18. Kundkrav			
18.1			
18.2			
18.3			
19. Kvalitet och tillförlitlighet			
19.1			
19.2			
19.3			

Appendix II

20. Lagerlivslängd 20.1 20.2 20.3				
21. Tillverkningskapacitet 21.1 21.2 21.3				
22. Tidsschema 22.1 22.2 22.3				
23. Testföreskrifter 23.1 23.2 23.3				
24. Säkerhet 24.1 24.2 24.3				
25. Interna begränsningar 25.1 25.2 25.3				
26. Marknadsvillkor 26.1 26.2 26.3				
27. Patent och litteratur 27.1 27.2 27.3				
28. Politiska och sociala konsekvenser 28.1 28.2 28.3				

Appendix III

Mötesprotokoll

Överskrift

Datum:
Tid:
Plats:

Sammankallat av:
Typ av möte:
Sekreterare:

Mötesdeltagare:

Dagordning:

-
-
-
-

Diskussion/Slutsatser:

Åtgärder:	Ansvarig:	Klart den:

Nästa möte:

Appendix IV

Chalmers	Relativ beslutsmatris:						
Utfärdare:			Skapad: 970924 Modifierad: 970924				Sid 1
Kriterier	**Alternativ**						
	Ref	**A**	**B**	**C**	**D**	**E**	**F**
Antal +		0	0	0	0	0	0
Antal 0		0	0	0	0	0	0
Antal -		0	0	0	0	0	0
Nettovärde		0	0	0	0	0	0
Rangordning							
Vidareutveckling							
Beslut							

Appendix V

☐ Konstruktions - FMEA
☐ Process - FMEA

FMEA - Failure Mode and Effect Analysis/Feleffektsanalys

Kund:	Utförd av:	Detaljnamn:	Detaljnummer:
Projektledare:	Anmärkning:	Datum:	Uppföljningsdatum:

Ordnings-nummer	Komponent	Funktion	Felkaraktäristik			Nuvarande tillstånd					Rekommenderade åtgärder
			Felmöjlighet	Feleffekt	Felorsak	Kontroll	F	A	U	Risktal	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	
										0	

F = Felsannolikhet A = Allvarlighetsgrad U = Upptäcktsgrad

Appendix VI

Rapportsynopsis

Sammanfattning

Handlingsalternativ, konsekvenser och rekommendation.

1. Företaget (kontext för produkten)

Kort företagshistorik, nuvarande inriktning, produkter, kunder, resurser (t.ex. personal, pengar, kontaktnät, utrustning och organisation), framtidsplaner och ambitioner, samt förutsättningar för det aktuella projektet

2. Produktidé

Beskrivning av produktidén och det behov som den avser att fylla eller problem som den avser lösa

3. Marknadsanalys

Kunder, användare, kundnytta, kundsegment, användarkrav, konkurrerande/alternativa lösningar (substitut), konkurrenter, konkurrentprodukter (pris, prestanda, för- och nackdelar, varianter, volymer), marknadskanaler, marknadsinstrument (annonsering, mässor, återförsäljare), underleverantörer, handelshinder (lagstiftning?).

4. Produktkoncept

Alternativa tekniska lösningar: idéutveckling! (Hur kan produktidéen realiseras? Alternativ?), teknisk analys av olika alternativ, definition och analys av kritiska tekniska problem (Vilka är de tekniska svårigheterna/flaskhalsarna/problemen? Lösningar?)

Teknisk verifiering av kritiska tekniska problem/lösningar(Hur kan vi lösa problemen? Fungerar dessa lösningar?)

Patent: analys av relevanta patent, och patentmöjligheter (Vad kan vara patenterbart? Vilka närliggande patent kan finnas? Hindrar några? Vad kan vi lära av tidigare patent? Vad bör vi överväga att patentera om vi når fungerande lösning?),

Överväganden om tillverkningsmetoder: produktionsanpassning(Hur ska produkten framställas? Alternativ? Krav som tillverkningssätten ställer på produkten?)

Kravspecifikation (baserad på tekniska möjligheter och marknadsanalysen). Vilka krav ska vi ställa på produkten? (Absolut nödvändiga, önskvärda), Definition av produktkoncept

5. Underlag för Prototyptillverkning

Ritningar, modeller och tekniska beskrivningar.

Appendix VI

6. Kommersiell bedömning

Förutsättningar för analysen:

Marknadsförutsättningar(Allmänekonomiska?Lagar? Förordningar? Marknadssystemet? Efterfrågan? Konkurrens/konkurrenter? Teknik? M.m.), Användningsområdets storlekMarknadspotential(=antal möjliga enheter per år idag?), Den nya produktens potential - produkten från användarens synpunkt(Hur många produkter av alla konkurrerande produkter säljs per år?), Spridningsförloppet(Hur många kan vi sälja över tiden?) Prissättning- prispolitik(Hur ska vi ta betalt? För vad? Hur mycket? När?)

Analysen:

Syftet med analysen är att totalbedöma ekonomin i projektet utifrån insamlad fakta och fattade beslut. Den kommersiella bedömningen är en syntes av ekonomi, teknik och marknadsförutsättningar. En typ av analysgång ges i kursmaterialet Tidig kommersiell bedömning av utvecklingsprojekt (SW2) som kan läggas in i exempelvis ett Excel-ark. Kassaflöde och återbetalningstiden vid fortsatt satsning är två viktiga bedömningskriterier för rekommendationerna.

7. Exploateringsplan

Kommersialiseringsmetod/strategi (Ska produkten kommersialiseras? Vilka alternativa sätt att kommersialisera den finns? Konsekvenser? Prioriterade alternativ? Val av metod/strategi!), Marknads- och säljplan, Varianter, Kundsegment, Säljkanaler, Marknadsföringskanaler, Kunderbjudande.Produktutvecklingsplan för fortsatt utveckling(Vad ska göras? Hur? När? Av vem? Resurser? Organisation? Styrning och uppföljning?)Kassaflödesanalys (Hur stort kommer kapitalbehovet att bli?)Finansieringsplan (Hur ska det fortsatta arbetet finansieras?)

8. Rekommendation

Handlingsalternativ, deras konsekvenser, risker, prioritering av alternativ, rekommendation för fortsättning, redogörelse för de viktigaste motiven till val av rekommendation, konsekvenser för företaget. Ni bör försöka formulera konkreta och handfasta rekommendationer.

9. Sammanfattande handlingsplan

De åtgärder som skall vidtas om projektet skall/skulle löpa vidare. Vilka aktiviteter skall genomföras? Av vem? När? I vilken ordning