Rauli Käppi

Organizational transformation from tailored software production to international Modified-Off-The-Shelf software business model

ABSTRACT

Käppi, Rauli Antero

Organizational transformation from tailored software production to international Modified-Off-The-Shelf software business model / Rauli Käppi

Jyväskylä: University of Jyväskylä, 2002.

151 pp.

Master's thesis

Software tailoring, also known as professional services has been a dominant phenomenon during the first decades of computing. During the past two decades a potentiated emphasis has been given to software products as a dominant way of delivering software. However, in European and Nordic context the development of successful international software products has proven to be challenging.

The objective of this study is to examine the software product business and in particular a product group of enterprise software, which must be customized before taken into use. The starting point of this study is a software tailoring company, as this is seen to be a very general situation for young product-oriented companies. This research has been carried out by using two different methods: firstly a theoretical literature review has been performed, secondly a constructive case study has been investigated.

Results of this study describe the development path for software companies with their customized enterprise software products in the international software market. At the same time the understanding of the expected challenges for small software tailoring companies targeting international software product business is developed. The consolidation of this type of knowledge is seen as one of the contributions of this study. Further results of the study are a stage growth model for companies starting software product business and practical suggestions regarding software product company focus, market segmentation and sales oriented company strategy.

KEYWORDS: Modified-Off-The-Shelf, software products, tailoring, enterprise software, startup, new product development, business process, internal process, process maturity.

TIIVISTELMÄ

Käppi, Rauli Antero

Räätälöinti, eli projektiliiketoiminta on ollut hallitseva ilmiö tietojenkäsittelyn ensimmäisinä vuosikymmeninä. Viimeisen kahden vuosikymmenen aikana ohjelmistojen jakelun painopiste on siirtynyt voimakkaasti kohti ohjelmistotuotteita, joista on tullut hallitseva ohjelmistojen jakelumuoto. Kuitenkin Euroopassa ja Pohjoismaissa menestyvien kansainvälisten ohjelmistotuotteiden kehittäminen on osoittautunut haastavaksi.

Tämän työn tavoitteena on tutkia ohjelmistotuoteliiketoimintaa ja erityisesti yrityksille suunnattujen ohjelmistotuotteiden ryhmää, jotka vaativat aina räätälöintiä ennen käyttöönottoa. Lähtökohtana tutkimuksella on projektiliiketoimintaa harjoittava yritys; tämän uskotaan olevan yleinen tilanne nuorten tuotesuuntautuneiden yritysten tapauksessa. Tämä tutkimus on suoritettu käyttäen kahta eri tutkimustapaa: ensin on suoritettu teoreettinen kirjallisuustutkimus, toiseksi on tehty konstruktiivinen tapaustutkimus.

Tutkimuksen tulokset kuvaavat ohjelmistoyritysten ja niiden räätälöitävien ohjelmistotuotteiden kehityspolkua kansainvälisillä ohjelmistomarkkinoilla. Samanaikaisesti kehitetään ymmärtämystä odotettavissa olevista haasteista. Tällaisen ymmärtämyksen tiivistäminen on yksi tämän tutkimuksen tuotos. Muita tuotoksia ovat malli jossa yrityksen kasvu kuvataan vaiheittain ohjelmistotuoteliiketoiminnan kontekstissa, sekä käytännön ehdotuksia liittyen yrityksen toiminnan fokukseen, markkinoiden segmentointiin ja myyntiorientoituneeseen yrityksen strategiaan.

Avainsanat: MOTS, ohjelmistotuotteeet, projektiliiketoiminta, yritysohjelmisto, startup, tuotekehitys, sisäinen prosessi, prosessien kehitysaste.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1 INTRODUCTION

One of the key areas of today's software industry is the production of software applications. This is an area where also Finnish government and companies have shown interest as a potential employer of their workforce. "The software business has the potential to grow to one of the most important industries also in Finland. This requires though new successful start-ups to emerge into the markets [...]." (Autere, Lamberg & Tarjanne 1999, 13). One of the factors in achieving this goal is the companies' ability to discover and fabricate software based functions and services, which are sellable to customers in a remunerating fashion. Whenever a software product concept invention happens in a software company, the difficult part is to find proof whether the invented software functionality and / or service is truly worth coding and how that software could be made successful in the commercial market.

The literature has plenty of material how large companies, such as Microsoft and Netscape conduct their software businesses in very large markets. What has been less investigated is how exactly should companies and new software products start their journey towards market success. At the same time it is commonly known that it is at large part the smaller companies that introduce innovative new software concepts. Product-oriented company types identified for this study are start-ups, venturing units of bigger corporations (above 1000 employees) and companies currently focusing on software tailoring business. The product-oriented companies are the main focus area for this research.

It seems that the society is placing high expectations for new software products and companies. On the other hand there seems to be little holistic, scientific knowledge of the smaller product oriented software companies (below 100 employees) and how the development process of a software company and its product(s) can be conducted from the start to the first commercially profitable position. Tailoring-oriented software companies are very much in a similar position with other product-oriented company types when they start to target product-oriented software business as their new, lucrative form of software business.

This type of setting is prone to create situations, where software companies may not be beforehand fully informed of all the requirements they are facing when making business and development decisions regarding their software products. This can be seen as a contingency factor both for the product to be developed and marketed as well as for the whole software company. Accordingly it can be argued that the ability to comprehend the international software product business and its requirements is a contributing factor for the success of software companies and their products.

This master's thesis studies and analyses two different software business models and the market approach of software product based business. It seems probable that Scandinavian (and even European) software companies have at least momentarily lost the majority of the software market to U.S. competitors (Autere et al. 1999, 9). A good example of this is the PC-market and the Windows operating system. This phenomenon emphasizes the importance of European corporate customers and their domain-specific needs, especially when examined from the perspective of economic growth in many European software companies. This area can be seen largely as a viable one, even for the small Scandinavian companies. As most of the corporate customers typically require integrated, partially tailored solutions, the area of Modified-Off-The-Shelf software seems to be a very important area for the future of European software product companies. To start examining the area, the different business models and other required terminology are defined first.

1.1   Software business models and related terms

Tailored software / -business model

A company is selling labour to produce new software for buyers (typically the buyer is another company), to fulfil their needs and expectations. Invoicing in this business model is typically based on hourly / weekly / monthly fee (or the work amount derived from a project phase) of the labour doing the task. Typically the projects can vary greatly in size, different technologies used and in complexity. Often the intellectual property rights (IPR) of the deliverables of the software project are owned by the buyer.

Another term for this software business type is professional services. In Finland this type of business is often referred as project business.

MOTS

Modified-Off-The-Shelf software. In their publication Rajala et al (Rajala, Rossi, Tuunainen & Korri 2001, 51) define the MOTS business mode to be similar to parameterised product with the definition of *"customisable product that can be tailored to a degree."* The most important points of variation of the software are in the design and installation phases in MOTS products. ERP-systems from SAP or Baan can be seen as examples. Typically there exists a certain level of tailoring needed on top of a MOTS software product, in order to make the whole service functional. The software is sold as a product and the IPR (Intellectual Property Rights) are maintained in the possession of the seller. Typically the customer purchases a licence for the product. Enterprise solutions is also a known term for this software business type.

COTS

Commercial-Off-The-Shelf software. Typical examples are Microsoft's Word or Adobe's Acrobat. A COTS software product does not need professional tailoring work when it is sold. The variation point of the software is controlled by the user selecting parameters from the installation menu. Typically this is a licence-based model of software business. In some cases the user may receive the licence for free to promote some other form of business model, such as the sales of server-side licences. This has happened in Internet browsers and streaming audio / video clients.

Productized, productization

("Tuotteistaminen" in Finnish). Regardless that this term is not known to many English dictionaries (but is occasionally used in the non-scientific literature), it is used in this thesis. In MOTS –context a software code / service is considered to be productized when it has at least the following qualities:

- the software is well documented for all the product's relevant user groups (e.g. user, administrator, buyer, seller)

- it is bug free, or at least the existing minor bugs are documented
- its usability is high
- its original problem solving ability is high.

All this contributes to the fact that a customer can purchase a well-productized software item, know beforehand what s/he is acquiring, and after installation have a fully functional software/service consistent with the documentation seen prior to the purchase. The process of accomplishing this degree of overall quality prior to sales phase is defined as productization. Please note that good productization can include also other components, such as pricing policies, other services to be combined with the product, etc.

Product concept

The Product Development & Management Association (PDMA) defines the concept for a product in their handbook as "a clearly written and possibly visual description of a new product idea, which includes its primary features and consumer benefits" (Rosenau, Griffin, Castellion & Anschuetz 1996, 520).

In the context of MOTS –business the consumer is usually a company, although end-users can also be involved within the buying company. This leads us to a somewhat broader definition: *A set of definitions defining both the overall functionality and clear added value of the product from the perspectives of all the actors involved with the product.*

Whole product

According to Geoffrey Moore (Moore 1998, 104) Theodore Levitt has presented the following Whole Product Model:
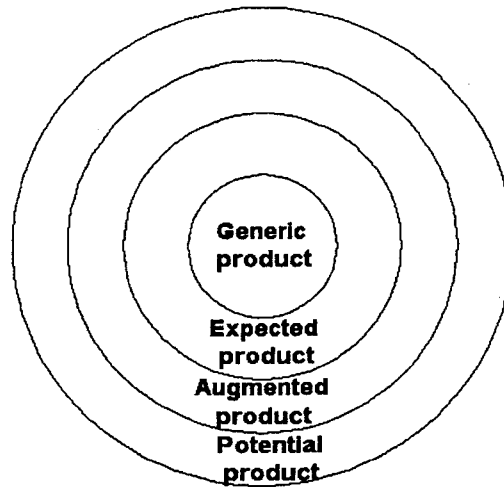
FIGURE 1 Whole Product Model

Moore gives an example of Levitt's model. In the Internet browser category the generic product would be the set of functions first made popular by Mosaic.

The expected product would include portability to each of the popular client platforms, including Unix and Macintosh. The augmented product would include plug-ins from third parties to provide additional features. The potential product would be the redefinition of the browser (possibly to include some or all of the functionalities created in the earlier phases as additional features).

Moore's own definition (Moore 1995, 21) of the whole product is more simplistic: *"The minimum set of products and services necessary to ensure that the target customer will achieve his or her compelling reason to buy"*. For this study of small MOTS product companies this latter definition seems sufficient, since in most cases the study does not cover very well established markets with many software generations of the same solution.

The ability for a software vendor to supply such a whole product is seen as a critical factor for having a satisfied customer.

R&D, PD and NPD

Organization's unit that is responsible for the technical development work of the software product. Research and Development unit in large corporations can be a unit, which is not responsible for specific products. In smaller product-oriented organizations the product related responsibilities typically exist and are categorized to be similar regardless of the naming of Product Development unit, New Product Development unit and Research and Development unit.

1.2   Research problem and the objectives of the study

The main goal of this study is to study MOTS product-orientation. The studied part of the life cycle is from the beginning of the product-decision until the first software product is in a good position to actually start targeting significant market opportunities commercially and technically. This path is approached chronologically to ensure different reader-types the opportunity to identify their own position in this development process. Three different real-life starting points are identified: A start-up company, a company already in tailoring business and a venture unit in an established corporation. For a closer study the company in tailoring business is selected, as the results are expected to be more general when neither of the extreme points are selected. Also the author has greater experience of this type of setting.

It is an assumption in this thesis, that the tailoring company's management adopts a growth strategy and that is one of the key motivators in entering the international product business in MOTS area where also the risks of failure and prior investments are greater than in the more easily controlled (often locally operated) tailoring area.

The research problem is: What kind of steps should the formerly tailoring-focused MOTS –company take to make its software product successful?

This research problem can be divided into sub-problems:

-    How should the product be developed in different phases?

- How do the internal processes develop during the period of productization and what competencies are required within the company during this same period?
- What should be the marketing and business strategy focus areas during this same period?

The results of this thesis should be extensively relevant to the management of primarily European start-up companies, companies exercising tailoring and even larger companies starting new ventures with new, innovative software products. The key common denominator is the fact that if a company is not already an established actor in a specific product field, it is facing the exact same path of negotiating with key customers with their new product concept (in order to make the product successful) regardless of the company's ownership. Customers need to accept the value proposition first and a technical development needs to take place in a similar manner regardless of the company type. Naturally, some of the tasks can be accomplished more easily in some of the three different company categories. If there are more resources and experience allocated for the task in a more established company, this can act as a contingency factor for the success of the developed product.

1.2.1 The structure of the study and research methods

First, the structure of this examination is presented. This is followed by the introduction of the used research methods. The high level approach to this study is to introduce the two domains (software tailoring and software product companies) under examination in this thesis. This is accomplished in the theoretical part forming the first half of this document. To validate and further strengthen the findings a case study is presented after the theoretical study. Final three chapters 8, 9, and 10 are based on the theoretical part and the case study. Those final chapters present the contributions of this examination with conclusions.

1.2.2 The structure of this thesis

The structure of this thesis is divided into ten following chapters:

First chapter, *Introduction* offers motivation in a larger context and identifies the research problem and definitions of relevant terminology.

Second chapter, *Different focus areas in software tailoring and software product business*, offers comparison between the examined domains. The purpose of this examination is to offer the reader an overview of the two domains.

Third chapter, *Overall description of a tailoring company,* offers a brief overall view of the environment in question. The presented information forms a holistic description of a company type, which acts as an accepted starting point of investigation in this thesis. The purpose of this chapter is to explore the tailoring area to create an understanding of this specific domain.

Fourth chapter, *Overall description of a product company* is described from several viewpoints to give a holistic view of the product oriented company, related managerial planning and company and product related strategies and targets.

Fifth chapter, *Fundamental Qualities of MOTS software business*, explores the intrinsic characteristics relevant to the specific product segment. The purpose of the chapter is to offer understanding of how and why this product type is different from other types of products. It is also suggested, that the product segment –specific qualities have a bearing to the organization and to the focus areas of the MOTS software company.

Sixth chapter, *From a tailoring company to a MOTS product software company*, is a chapter describing the rationale and motivation for a tailoring company's management to implement the transformation process from one software business domain to another

Seventh chapter, *Case study,* offers a reconstructive, two-year longitudinal study of one company's two different MOTS software products and how those have behaved in the international software market. The purpose of this chapter is to offer concrete, realistic evidence of the domain and its behaviour.

Eight chapter, *Theoretical contributions - Designing the framework for the MOTS product company domain,* analyses the theory presented in the earlier chapters with the case presented in the previous chapter. The motivation for this chapter is to form

synthesis between the case and the theory. The domain of examination is relatively scantly covered in the academic literature. This suggests as a significant contribution to be the scientific discourse in the area of emerging software business discipline. A *stage growth model from tailoring to MOTS software product business* is presented and discussed. The model offers a framework for holistic understanding of the transformation process under examination in this thesis.

Ninth chapter, *Practical contributions,* offers further conclusions and practical suggestions based on the findings of this study.

Tenth chapter, *Conclusions,* recapitulates the findings of this study. Final, high level discussion of the area and its academic research is presented.

1.2.3   Research methods used

The study presented here is a qualitative study, which consist of theoretical examination and of (re-) constructive case study research. The resulting artefacts are practical suggestions regarding the studied domain and an abstract development model for companies starting software product business. According to March and Smith the research has elements of theory building, since a model can be considered as theory building (March & Smith 1995, 256).

Regarding the constructive research this study has clear elements of action research. However, one criterion is missing for this study to be regarded as action science. In action science the formulation of the research problem is carried out prior to the actual study. The researcher then works on one hand to solve the problem and on the other hand to create and acquire scientific information related to solving of the problem. In addition there should be an immediate feedback cycle from research to the action during the problem solving (Järvinen & Järvinen 1996, 79). In this thesis the formulation of the research problem has been carried out only after the action has taken place. This is why this study can be characterized as retrospective and reconstructive.

The process of building a theory from case study research according to Eisenhardt (Eisenhardt 1989) includes the following steps to be taken: Getting started (defining the

research question), Selecting cases, Crafting instruments and protocols (for data collection), Entering the field, Analysing data, Sharpening hypotheses, Enfolding literature, and Reaching closure.

In the beginning of this study the case had just been concluded, which allowed the strategy of collecting evidence from the selected period and analysing and interpreting it. Carrying out this type of research can be seen as two-folded: On the other hand the fact that the author has worked and carried out a major role in the examined company can cause bias to the study. On the other hand by working closely with the case in question very much insight can be gained compared to an interview, where the interviewee can already filter out unpleasant issues. To minimize the effects of the bias and to sustain an objective view to the area of research, a data triangulation method has been used. Miettinen (Miettinen 1998, 44) suggests that data triangulation "involves checking inferences drawn from one set of data using other data concerning the same phenomenon and independent from the first set." In this examination the theoretical literature review has been exercised to gain insight and to be able to establish validity of the examined phenomenon. Further sources of compared data have been the communication in the company, and the published documentation presented by the studied company itself. It is notable, that carrying out the literature review can be seen as an essential process to be able to produce the contributions presented in this study.

Comparing to Eisenhardt's steps, the research question has been crafted in the beginning as suggested. The case selection has is somewhat different, since the case is a reconstruction of the past as is the data set collected. One cannot really say that intruments and protocols would have been crafted for data collection in this study. As the theoretical literature study has been carried out after the research question has been formed, this part of the study is similar to entering the field in Eisenhardt's suggestion. The retrieved data has been analyzed; hypotheses and assumptions of the studied field have been sharpened during the discourse of this study. Finally the literature has offered essential components to reach the results and closure of this study. This process has served as an invaluable part of creating an enhanced level of understanding of the domain for the author. Hopefully this academic work is able to pass a similar experience of insight to the reader of this document.

1.3   Limitations of the study

In this thesis the funding for the software company / product is not covered in detail. For readers who are interested in this area there already exists material, such as Venture Capitalist (VC) –literature where these issues are covered discussed. A similar issue in existing tailoring and corporate companies is typically a matter of locating existing resources to product development through some decision-making process within the company. The focus of this study is to explore the development of the product and the company in a situation where the funding-decisions are already made, although some clarification to the overall rationale of making such decisions is offered.

Another limitation of this study is that the issues regarding time-to-market (time-to-completion) for the software product releases are not handled in detail in this study. The company's ability in terms of time-to-market may be a critical factor for an established company producing new software releases. Even in a case of an undeveloped market where the competitive forces are smaller, the time-to-market may prove to be an important factor also.

The different reader types of this research are identified as tailoring companies, start-ups and venturing units of bigger corporations. However, the validation of the results of the study is not attempted regarding the start-ups and venturing units, also the results are not developed / validated with the COTS environment.

Many of the conclusions are based on theoretical literature study and the case analysed. Since there is only a limited amount of literature available regarding the examined area, and almost all of the empirical findings concern a single company, the validity of the conclusions would be helped by a study concerning multiple companies. This can be seen as a task for future software business research.

2 FOCUS AREAS IN SOFTWARE TAILORING AND SOFTWARE PRODUCT BUSINESS

The software industry in itself is a vast area, which is difficult to examine in detail as a whole due to its variety in nature. Therefore it is useful to categorize the area into more homogenised areas for examination. Käkölä notes (Käkölä 2002) "The industry can be divided into five major segments: professional software services (planning, building, integrating and maintaining customized software systems), enterprise solutions (relying on both products and professional services that adapt the products for customer needs), packaged mass-market software, internet-based applications rented by Application Service Providers, and embedded software including services."

The professional services business is referred as tailoring business in this thesis. This thesis also covers the area of MOTS products, which Käkölä refers as enterprise solutions. The focus of this thesis is to examine the area of such MOTS products, which can at first be designed and implemented within the limits of small and medium-sized product organizations. To start the investigation relevant to this thesis the areas of tailoring and software products in general are first examined.

The definitions of a software product business and software tailoring businesses are often confused, since the overall technical and business perspectives may at first glance seem similar. Sometimes also the terminology is confusing as the tailoring companies are typically very project oriented, and also the products are implemented in projects. Product business is compared to tailored professional services business in the following table (Hoch, Roeding, Purkert, Lindner & Müller 1999, 46).

TABLE 1 Software product business compared to tailored professional services

	PROFESSIONAL SERVICES	PRODUCT BUSINESS
MARGINAL COST	Almost constant	Almost zero
MARKET STUCTURE	Highly fragmented	Highly globalized
REGIONAL APPEARANCE	Mainly regional, with increasing tendency to globalisation	Higly globalized
CUSTOMER RELATIONSHIP	One to one	One to few, one to many
MOST IMPORTANT NUMBER TO WATCH	Capacity utilization rate	Market share (installed base)
RELEVANCE OF MANAGEMENT AREAS	1. Human resources 2. Software development 3. Marketing and sales 4. Strategy	1. Strategy 2. Marketing and sales 3. Human resources 4. Software development

As described in the table the most important number to watch for professional services is the capacity utilization rate. It can be concluded that if the clear target of the management is to keep all the employees utilized in billable projects, it does not suggest a very complex strategic or business models to be used. In fact the business model of a professional services company can be characterized as simplistic.

For product business the table suggests the strategy as the most important issue for the management. This is a key point, since the products and their business models are derived from the overall strategy. All these issues are management's tasks. Here are some issues that make the work of management in a product focused company complex:

- There is no pre-defined order from a single customer.

- There is no pre-defined, single environment for the solution.

- There is no single customer to accept the suggested specifications of the software.

- Selecting and having sales partners to cooperate with the sales and distribution of multiple licences is a continuous task.

- Customer(s) need to be kept constantly well connected with the software company, even through partners acting as intermediaries.

- Full documentation of the product must be offered to the customer prior to sales and delivery – not after the project.

Comparing to (Hoch et al. 1999, 34-38) tailored software services seem clearly easier from the management and strategy perspective than the software product business.

The problem field for a product-oriented company's management is complex. To examine the area prior to actual product releases the following thinking is suggested, adapted from (Äijö 2001).

TABLE 2 Focus areas for product company's management

	Function	
Perspective	**Internal**	**External**
Business	**Business model** - What are the needs the company is aiming to satisfy, what kind of products the company wants to sell? - What kind of customers are wanted - How do the customers want to be serviced - What is the revenue model of the company	**Model of operations** - What is done internally and what is outsourced - Who are the partners, what kind of relationships are created with partners and how are those maintained (strategic alliances and partnerships) - Which models of working are used (contracting, licensing, franchising, capital investment, company acquisition, fusion.
Value creation	**Internal value chain** - Support functions (general management, organizing, funding and HR - Value creating functions, such as procurement, technology and product development, internal and external logistics, production, marketing and customer services	**External value network** - Company's position and role for example in the context of value chain of the particular business area - The structure of companies and the network of companies in the area - The relationships between the companies in the area

It is notable in the table that even when attempting to form a holistic vision of the company prior to entering the market; this may prove to be too large of a task at one time. Instead the management may be forced to a situation where some of these issues need constant attention and can only be crafted after gaining additional knowledge of the field the company is working on.

Käkölä (Käkölä 2001) notes that in volatile environments markets are typically equivocal, software companies cannot search for and find all the information that would help decrease uncertainty and risks in business development because it is extremely hard to foresee what kind of changes there will be in their environment even in the near future. In such contexts, software products and services can seldom, if ever, be

accurately defined at the outset of the development process. Competitiveness therefore depends heavily on software companies' abilities to develop business and application domain competencies, software engineering and process competencies, business models, organizational and information systems designs, and product development, marketing, and distribution strategies that maintain dynamic balancing between exploitation and exploration and help realigning planning and execution of actions in time to the maximum extent.

Crafting the focus and strategy for a product-oriented company is not easy and it is difficult to create a holistic, detailed vision of what, to whom and how the company is to sell its products. This does not however reduce the importance of creating such a vision as soon as possible. It is in no way less costly to "wait and see" as a software company, spending resources on company's product-driven functions without a clear, guiding focus for the whole company. In the worst case this type of situation can prevent the formation of the domain specific knowledge (both tacit and explicit), if there doesn't exist a clearly focused area to learn from for the company's personnel. This type of situation grows acute the moment the software company makes decisions of spending resources on software product domain.

Strong emphasis on management's importance is presented by (Autere et al. 1999, 31-32). According to them the management of a company is the key to all companies' success, including software product companies. Furthermore they note that often the key to success is to hire an experienced director to run the company. However utilizing experienced directors in Finnish software product business is difficult. They list the reasons to be the scarce resources in the field and the domestic nature of the available experience. Their suggestion to remedy the situation is to acquire experienced directors either from other business areas or from other countries.

Considering tailoring business' nature of selling labour there are several tasks to be accomplished by the management of a tailoring company prior to entering the field of MOTS products. A very large portion of that work is allocated to the management before entering the field. This means that in the beginning much of the work to be done is not necessarily technical, it's managerial. In (Rajala et al. 2001, 33) the following is noted: "The implication of corporate strategy and business strategy to software vendor's

applied business mode is a relevant issue if we consider business model to follow the goal setting and design of both corporate and business strategies."

"For example if a software venture has adopted a growth strategy, its business model is likely to be product-oriented. If the strategy is to focus on some specific domain of knowledge instead, the business model relies on specific competencies, including management competencies of the management team" (Ibid).

The challenge with many small Finnish tailoring companies is that they do not have the resources or contacts to hire top management. This can very easily contribute to a situation where the whole process of evaluating the business environment (including technical perspective) is not done professionally. Failing to do so jeopardises the whole interrelationships of company strategy, product focus derived from that strategy and the business model created using products and strategy as factors. There is very little evidence in the literature that any single unit in the company (such as new product development, marketing, sales, customer service) could remedy the situation on their own, if the basic strategy and focus is designed poorly.

Considering that company is able to execute the analysis of its surroundings and its own focus successfully it is possible to proceed to planning the actual product concepts for certain business / customer segments. For understanding the field of consideration for a new software product and its business environment, the following, careful examination is offered, adapted from (Davis 1996, 36)

Business synergy:

1. If the market is one that is new to the company, are its management skills, talents, and experience applicable, or will it have to learn a new business?

2. In this market, can the company build on its technological strengths to gain a product advantage, or will it have to acquire and learn new technology?

3. Is there marketing synergy? Can the company make use of its existing sales force and trade relationships, or will new channels have to be developed?

Market attractiveness:

1. Is the market a large one?

2. Has it been growing over the past few years?

3. Are there many competitors in this market?

4. Who are the competitors?

5. Is there a dominant competitor?

6. What is the pricing of the leading competitors?

7. What is their cost structure?

8. What are their sales, advertising, and promotion strategies and practices?

9. How have they reacted in the past to new competitors?

10. What are their technologies, and are they protected?

A sufficient understanding of the above *on the product level* is required in order for the management of the software company to be capable of making rational decisions of entering such an international field. This type of information is certainly not obtained automatically. The management of the software company should allocate proper resources, at minimum several man-weeks of labour in order to explore these issues if the area is not already well understood. All this work cannot be carried out by the more inexperienced personnel, since analysing and investigating these issues cannot be done by a person who is not experienced enough to know what s/he is looking for. This implies heavy involvement of the more senior management during this whole process. Some of the market-related questions may not be relevant to a software product, which is offering a new service not introduced before. However, even those cases the state of things should be clearly understood and communicated within the management to have a common understanding.

It is clearly visible that matters such as product concept and product development, company structure, marketing efforts, and selected customer segment are not easy to manage simultaneously, especially if they are globally distributed and the market is

creating new requirements for the product over time, thus changing the situation as time passes.

It seems evident that a software company focusing on tailoring does not have the required competencies, especially managerial resources and competencies developed during their domestic tailoring business, which would be required in the international software product business domain. It can be questioned whether even the domestic software product experience is sufficient for a successful international operation.

In many aspects the internal requirements for transformation of the company are substantial. Analysing only the surface, program code, used technologies or the fact that similar titles exist in both type of companies (such as a developer, web-designer, project manager, CEO) can give a very misleading picture of the level of transformation required to be successful in the product market. One direction of proceeding can of course be the option of starting first with a small level of internationalization, first the domestic market and then for example the Scandinavian markets. This type of approach can reduce the risk and give the organization time to adapt slowly to the change. However, in some products this may be difficult due to the small size of a single niche in Scandinavia.

It is recommendable that if there aren't any internationally experienced personnel already within the software company, a good amount of outside help should be acquired to assure that all the different requirements of the international software market are understood and accounted for. Failing to acquire this type of knowledge can easily result to a situation where the conclusions regarding the international software product business are based on domestic market experiences and assumptions, which may have only vague resemblance to the international markets. This type of external help is offered by different government agencies, consultants and by newly hired personnel.

To understand fully the starting point of a tailoring company and the target to be reached as a MOTS software product company, the next chapters examine more closely these two different company types and their special qualities.

3 OVERALL DESCRIPTION OF A TAILORING COMPANY

Tailoring is the starting point of many small IT-companies' businesses (Warsta 2001, 35). Given this as a starting point of real-life companies, it is interesting why the success rate of small product companies is not as good as one could expect, especially given the emphasis on IT in today's society. Tailoring companies do have benefits compared to start-ups. They already have some personnel and competencies in some technical areas, if their development process has been practiced for some time, even that may have reached some level of maturity. Therefore one could expect them to have good opportunities to succeed in the new product development area. However, the success of the software product-oriented companies should be investigated and verified. There exists an assumption within this thesis that some factors of international software product business may have been underestimated by some of the domestic software business companies. In this context the definition of software product company's success is not fulfilled by the mere successful issuance / sales of the stock of the software company, but instead the remunerating sales of the products of the software company.

In the following subchapters the tailoring company's business model, internal competitive factors, a project as a billing mechanism and the internal competencies developed are examined. This is done briefly to allow the reader to proceed to the actual product-related chapters.

3.1 Business model

Project-based business model companies rarely sell licences; in some cases even the price of the finalized solution is not fixed in the beginning of the project. Instead the business model supports mainly fixed pricing of a single worker / phase of the project. The basic objective is to sell as many billable units to a customer as possible. The next figure takes a deeper look to the business level interaction with a customer in the case of the tailoring company.
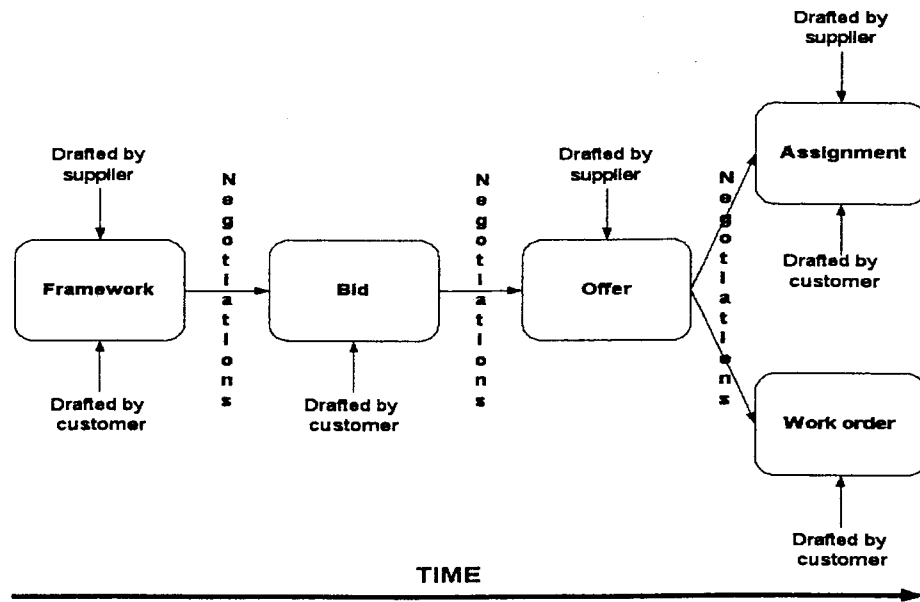
FIGURE 2 The assignment and order contract drafting in tailoring according to Warsta

Juhani Warsta (Warsta 2001, 146) suggests that a framework contract is a central part of the tailored software business as the customer and supplier aspire to a long-standing relationship with mutual commitment and understanding. A large part of the work is done in negotiations where relevant issues are communicated between the parties to enable a common understanding and all documents to be drafted. After receiving a project bid from the customer the software company prepares an offer. The acceptance is documented in the form of a work order for the approved offer. Another form of acceptance is an assignment contract, which further details the work to be done; both parties sign the assignment contract.

It is notable that a framework contract may be more suitable for larger tailoring companies and they may not be very common among the smaller companies, which may be more project-by-project oriented. The motivation to bill any unforeseen additional work from the customer exists. If this is not prevented by the contracts made between the two parties, the tailoring-company responsible for doing the additional work always seeks first the acceptance from the customer for the specified amount of

new billable work. Hence the business risk is by default largely on the shoulders of the buying customer. Naturally these issues are negotiable, but by default it is very rare for the project company to carry the whole business risk for the customer. The project companies have therefore a very different marketing approach to acquire their customers compared to the product companies. Instead of trying to convince the customers of the superiority of some technical feature, they must market themselves, their CMM-level X processes, their overall technical knowledge and other issues related to their competence, hence not any specific piece of software.

3.2   Competence management

Since the business model of the tailoring company is relatively simple the need for a large amount of management in a tailoring company is not substantial. A typical Finnish small / medium-sized tailoring company may have a relatively high number of project management personnel, but the need for a large amount of other management is minimum. Naturally, in the large, corporate sector the overall situation may differ. For even larger tailoring companies the experience regarding international business may be small, since typically all the tailoring projects are done domestically, as can be noted in (Autere et al. 1999). All this indicates that even when the need for management resources and management skills is relatively high in software product business, it is hard to see those requirements be fulfilled by a tailoring company's existing management in general. In the lower levels of the organization the situation may be similar.

Another matter of interest is the software developers. Their task is to design and implement the promised software solution. To accomplish that the project company must adopt a rather wide variety of skills in different technological areas such as different programming languages, different interfaces (such as Web, WAP, SMS, Windows-clients), different database vendors, etc. At this point the economies of scale start to affect. In order for the company to cover such a wide area a bigger size is an asset since there is only a limited number of technologies a single worker can be an expert in. Another possibility for a smaller company is to utilize the economies of scope

by focusing on certain area, for example customer segment. By doing this the project company can limit the amount of technologies thereby creating a credible set of competencies "more cheaply" for their customer group's needs. These economies of scale and scope can and have been combined to create a more credible whole, but that is a matter of another discussion.

As these competence areas and technical expertise are very commonly utilized in the billable projects, the project entity itself is examined further to understand the tailoring company's internal processes more deeply. This also contributes to the understanding of the tailoring company existing as a starting point for the product business.

3.3   Project overview

As described earlier, the project company tries to sell its labour. In order to accomplish this, the company (and its salespersons) actively seeks customers having software-tailoring needs. When a customer is identified, the amount of work is estimated and a quotation for the work amount is passed to the customer.

If not done earlier during the quotation phase, the project must typically start by analysing the environment, defining the needed functionalities and required technologies.
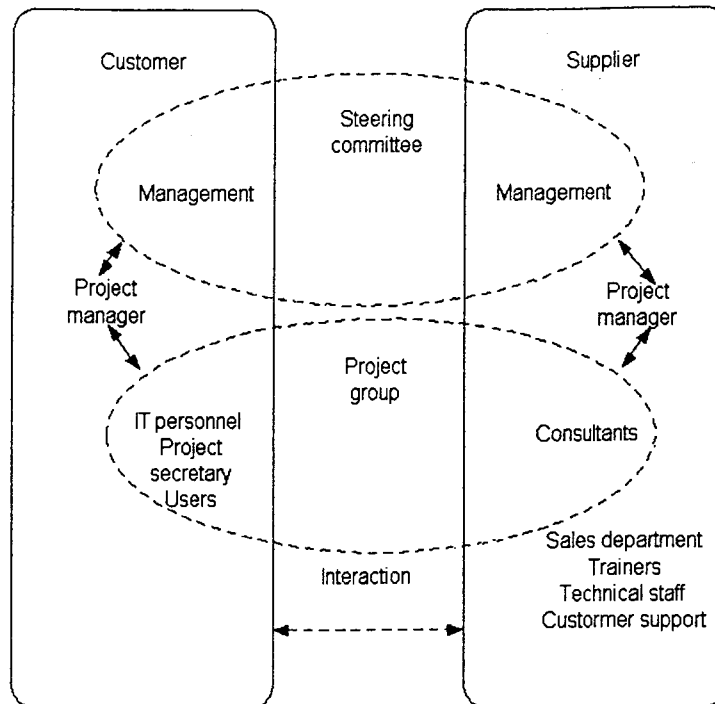
FIGURE 3 Organization of tailored software projects (Tähtinen 1999,139)

FIGURE 3 represents organization of a large tailoring company interacting with a customer company. When the tailoring company / project is small, the steering committee does not necessarily exist. Instead the issues may be handled between assigned managers from both companies. In this case typically also the work done in the project is more the responsibility of the tailoring company, thus causing the oval of Project group to shift to right. Signs of bigger project are also consultants, who typically don't participate in smaller tailoring projects.

One view on the project companies' ability to be competitive can be seen as their ability to execute the projects with good quality and speed, which in turn can be reflected with their quality of internal processes. This is supported by the TABLE 1 Software product business compared to tailored professional services on page 21 where the two most important issues for the management of a tailoring company are human resources and software development.

3.4   The competencies developed within a tailoring company

The fundamental suggestion here is: Company's personnel increase their competencies in the areas they work on. Other areas of competence do not get developed if there isn't a significant amount of work done in that area. So what are the areas small tailoring companies get to practice the most? First and foremost the technical skills is the one area, which after a while is to be developed as the strong suit for the tailoring company. If the company is involved with larger projects, also the management skills of different projects could start to grow as their strong area. Most of the activities are domestically focused; so international business environment is not by nature a strong area for the tailoring company (Hoch et al. 1999, 46).

In the tailoring company, very little work has been done with extensive documentation for different user groups, software-specific marketing or following the competitors of similar technical solutions. There simply aren't any processes established within the company, which could secure that these issues could be dealt with. Naturally this is due to a fact that there is also a very little need for them.

Looking at the direction of product business the basic project management skills of a tailoring company may be sufficient for product business (in very small ad hoc companies even this cannot be taken for granted). However the understanding what exactly is needed in the technical and documentation level may be misleading if the only experience is from the tailoring field. Issues such as future solution scalability, performance with n different hardware settings, different supported environments etc. are not necessarily on the requirements list of a single project focusing on a single given environment, not to mention the documentation of these issues.

In comparison, the qualities of a product-focused company are described in the next chapter.

4 OVERALL DESCRIPTION OF A PRODUCT COMPANY

To succeed in the international software product market it is essential that (at least) the upper management of the company is able to form and maintain a clear focus of what, to whom and how the company is going to sell. These issues can be seen as core parts of the overall strategy of the company. If any of these very basic issues is missing, it is very difficult to perceive the company reaching a mature and profitable state of existence.

In chapter 2 a list of product-related issues for consideration was presented, adopted from Davis. As that list strives to seek for basic information of the considered product area, a more holistic thinking process is beneficial for the assessment of the software company's direction and focus. The following presents a company-level view how the software product company should focus themselves. This view relates very much to the external indications and their effect on the software company's direction.

Vision, Mission, Value proposition

External Analysis   Internal Analysis

Threats   Opportunities   Strenghts   Weaknesses

Strategic Change   TQM, BPR
Management

Matching Process
Market Intelligence and Research   Technology and Marketing Match

Market Segmentation

Buyer behaviour   Targeting Individual Accounts   Relationship marketing

Customer Satisfaction and Loyalty

Product   Quality   Pricing   Communication   Sales Force   Service   Distribution and
Strategy   Strategy   Strategy   Strategy   Strategy   Strategy   Logistics Strategy

Deployment and Implementation

Monitoring and Improvement

FIGURE 4 The market-driven planning framework (Räsänen 2001)

Proceeding from the top of the figure the management may already have an idea how the company could operate in the product market in a remunerating way. This first idea needs analysis and thought before it is used in real-life investments and activities. The analysis is divided into two parts, external and internal. The most promising area for future product business is where both the external opportunities are perceived, and the company's own strengths exist. For overcoming market threats strategic change management, and for company weaknesses total quality management and adaptation of best practices are suggested.

Matching the software product concept is also a matter of doing additional market research and analysing the suggested technology and the company's ability to market it. If the software product business still seems viable, the management can proceed to

segmenting the market to find out who are the exact company customers the product should be targeted to. This work includes exploring and analysing this segment's buyer behaviour and assessing the possibilities of relationship marketing. After the segment's behaviour is understood the management can approach selected customers in the selected segment. These discussions should give a fair comprehension of the whole situation from a commercial standpoint. When looking further the management of the software company should also be able to form a view of what level the customer's satisfaction and loyalty will be after purchasing the product. Typically in MOTS products the loyalty can be high due to the tailoring work involved, which presents an extra hindrance in changing the vendor.

After planning and finding out these issues the software company can proceed to crafting a more detailed strategy regarding the seven different areas described in the figure. After a holistic picture of the product company's functionality has been formed, this plan should be implemented in practice. Finally the implementation should be monitored and the plan be adjusted accordingly when real-life indications suggest a modified approach.

4.1   Focus on the areas of excellence

To be successful in a given area of business the company is suggested to focus on one of the three operational strategies. According to Käkölä (Käkölä 2002) different strategies are "operational excellence, customer intimacy, and product (or service) innovativeness. Operationally excellent organizations have superior quality and programmer productivity, low costs, and a stable and focused product or service, which is delivered excellently and at a competitive price."

"Customer intimate organizations offer complete, often customized solutions. They seek long-term relationships to understand the underlying value networks of their customers well" (Ibid).

[...] "Product-innovative organizations typically target mass-markets and compete by trying to launch new products faster than their competitors, innovate features markets

are most willing to pay for, and move rapidly to new products and uncontested markets in order to generate maximum revenue streams. They operate within a context of rapid technological advances, short product life cycles, organizational transitions, and competitive, international markets. Product innovation strategy dominates all software industry segments except for the professional software services segment" (tailoring segment) (Ibid).

[...] Software businesses need to crystallize unique strategies by properly balancing aspects of all three strategic alternatives and leveraging partnering optimally. Excelling in one alternative is necessary but not sufficient for success. They also need to implement and dynamically adjust such strategies that maintain a proper balance between short-term efficiency and long-term effectiveness (Ibid).

Discussing of the two presented strategy models (Räsänen and Käkölä): the planning presented in the previous FIGURE 4 The market-driven planning framework (Räsänen 2001) may need to come first for the software company to be able to find a product and business area to begin with. As the FIGURE 4 suggests 7 different, focused strategies to be developed, these are the focal points that should reflect some of the three operational strategies suggested by Käkölä.

In the case of MOTS product business for a tailoring company, the product concept(s) may consist mainly of good, innovative ideas, which can be encoded into software (and some may have already been implemented). Hence, the selected operational strategy should first have emphasis on the innovative and customer-intimate strategies, which then should be reflected in the 7 different strategy focus areas presented by Räsänen. Innovative strategy is required because it is very difficult to perceive the demanding enterprise customers buying software products from a new, untrustworthy positioned product company, if the product does not even have any new, added value to offer. This type comparison and situation is unacceptable for the new product company. Compare to (Cusumano & Yoffie 1998, 222-233) where Netscape had a very strong emphasis on innovative new features in the beginning to gain popularity for their software browser product. The customer-intimacy is required to understand the customer's ability to utilize the suggested enterprise software solution and to thus create added value for the customer company. If the software company is able to succeed in the selected product

segment, emphasis on the operational excellence will be required, as the competition and increased customer requirements are starting to effect. Again, this can be compared to Microsoft's operating system and Netscape's browser, which changed their emphasis towards quality and the ability to deliver the products promptly (operational excellence) as they proceeded further from their first product releases (Cusumano et al. 1998, 222-238). Even in a described, more mature product situation, the MOTS product company's core competence must remain in the customer-intimate sector, at least to the extent that the software company is able to receive information, understand and codify the requirements presented by the customer companies' environments. Failing in this will result to the limited ability to fulfil the future domain-specific requirements. The level of product-innovativeness may gradually reduce compared to competitors who are able to copy many of the innovative ideas. Even then the new internal innovations should are always of value to the MOTS software company. It must be added that the emphasis on innovation is heavily interrelated with the product area in question and there can be software product areas, where the innovativeness of the products is less important than understanding the customer's value creation.

If the presented synthesis of the two strategy models is accepted, the indication is that the MOTS software company may need always to adjust its 7 strategy focus areas when the company and its products grow more mature. This is caused by the transition from product-innovative and customer-intimate focus to a more operationally excellent direction. As the 7 strategy focus areas are first implemented to reflect the first setting, those 7 areas should later be adjusted to reflect the setting in a more mature situation. As the existence of more than a few years is the goal of any software company, it is important to understand the more mature internal development processes of the products. This is covered in the next subchapter.

4.2   Maturing the internal development process

Paul Trott (Trott 1998, 141) presents a view of the process of creating new products or new versions of the old ones. This view is perhaps more bound to the internal functionality and processes of the software company. As the higher level issues have

been mostly solved, the management of the software company can move on to a more detailed level of planning. The very high level issues have been shown in TABLE 2 Focus areas for product company's management on page 23 and in the previous FIGURE 4 concerning company-level planning. In the following figure a more mature, generic internal target process is presented of how the software company can actually operate to create added value effectively (Trott 1998, 141).
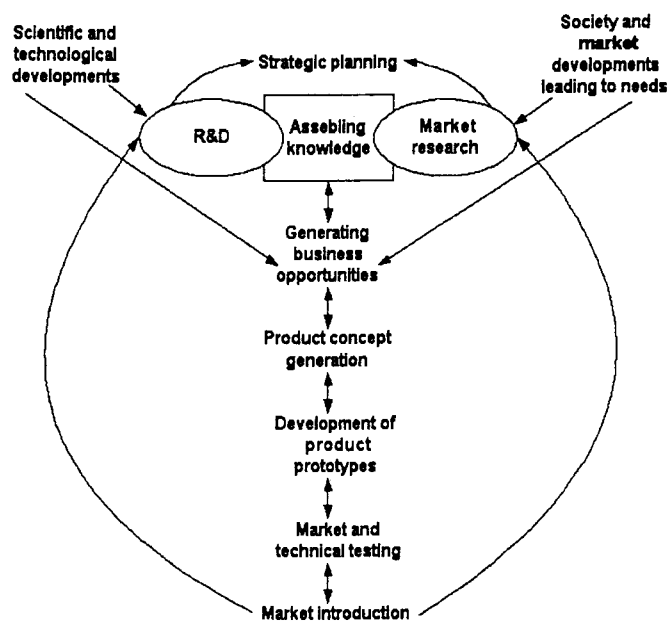
FIGURE 5 New Product Development (NPD) as a series of linked activities

This figure describes a very mature process, which is typically utilized in more established companies by possibly different teams handling their own expertise areas. In small companies the same phases may be carried out by a smaller group of people responsible for multiple tasks.

On top of the figure the strategic planning is (again) emphasised. Secondly there is the area of Research and Development (R&D), Market research and *assembling knowledge*. Trott (Trott 1998, 141) emphasizes the assembling of knowledge as being a key component of organization's ability to create a base and common understanding of the

perceived environment and its possibilities. Based on this acquired knowledge the company can proceed to Generating business opportunities-phase. The quality of carrying out prior phases has its importance. If the knowledge of the domain is weak, the perceived opportunities may prove to be false too often. On the contrary, if there is solid knowledge of a certain domain, the business opportunities perceived very often exist also in reality.

The next task is to form real-life product concepts to serve the area where the business opportunities are seen. If a product concept entity is seen to be of good quality, it may be passed on as a development project where a prototype is created. After passing technical testing and enhanced with added requirements indicated by customer feedback, it may be developed to a fully implemented product and introduced to the selected market segment. The lessons learned during and after the time of the product launch can in turn be used when planning a new version of a software product or new complementary products to the area in question.

After the product launch the software product should start following the framework to be presented in chapter 4.6, The suggested product sales and marketing development according to Geoffrey Moore on page 50. New versions of the software product normally need to be developed when the product is advancing to subsequent phases of Moore's framework. Each major new release of the software can use the same NPD model when being crafted, every time giving emphasis to different phases of the model on a contingency basis. Naturally, when the product matures, it may not be necessary to thoroughly analyse the Generating business opportunities-phase every time. At this point of a software product's life cycle more emphasis and resources can be put to the later phases of the product's version development.

4.3   Company maturity and release development

Carmel (Carmel 1994, 502) suggests an interesting growth model for the development practises of young software product companies (start-ups). The following table is adapted from his work to provide a high level understanding of the situation in different phases of the software company.

TABLE 3 A generalized stage growth model of software product companies

	Product concept → first release	Release 1.0 finalized → first steps in the market	10 employees A transition phase	20 employees Rapid growth
Development methodology	Ad hoc	Introduction of configuration management & error sheets	Transition	Elements of formal methodology begin
Organization	Formation of a small core development team	Non-development staff is added	Transition	Additional development staff is added
Quality assurance	Minimal	Some	Transition	Formalization begins

In the first phase a small core developer team emerges. This small (2-6 persons) team creates the first release of the software with ad hoc methodology and virtually no formal quality assurance mechanisms. After the first software release is created, the company hires personnel to handle tasks such as sales, marketing and product management. More formal methods start to be adapted as the company grows and the adopted ad hoc mechanisms start leading to chaos with added personnel. The functioning of the software company nears formally the situation described in FIGURE 5 New Product Development (NPD) as a series of linked activities, as the company moves to the right on this stage growth model. As the former table has been a presentation of how the software company matures, the following (again) offers a more precise view on the company's internal target processes. The assumption of the following FIGURE 6 is that a very mature process has been reached and all the required competencies of a product company have been gained.

A new release of an existing software product can be created using the following mechanism. The assumption here is that the company has already reached (and preferably passed) the right side of the above TABLE 3. Further assumptions are that

the company has already focused itself and gained sufficient knowledge of the particular MOTS software domain. After this type of positioning is reached, it is possible to utilize a more domain-centric focus in internal processes. It is also interesting to compare this figure with the FIGURE 5 New Product Development (NPD) as a series of linked activities on page 39, where the domain-specific focus is not strongly reflected, but instead a certain amount of searching is allowed.
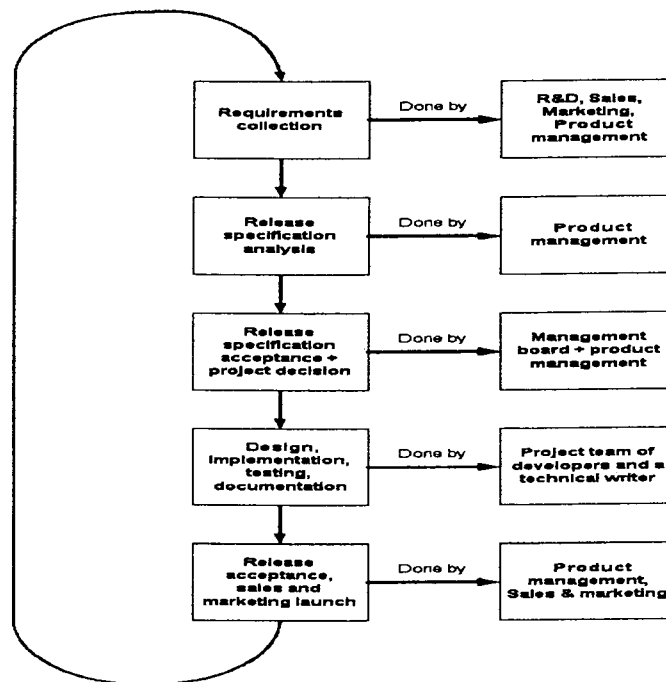
FIGURE 6 Release development process with different actors

In a typical stage-gate model all these phases include gates of approval which are used as tools for the management to control the process and to be able to make a discontinuance decision, if the project / product is not seen developable for some reason (Cooper 1996, 475-481).

It is notable that the emphasis and exact division of related strategy tasks may differ between the upper management and the product management from company to company. As long as there are clear responsibility areas within the company and these

are in parallel of the tasks defined in the company's internal process, it may be somewhat irrelevant how these tasks are divided between the two parties.

Beginning from the top, the ideas, inventions, customer requirements etc. are collected. This task can be seen as active all throughout the process. Primarily the product management (which in a case of a small company and one specific product typically constitutes a single product manager) is responsible for gathering the data arriving from different sources. Sales and marketing have an important role as they are mostly connected with the customers.

After the requirements have been collected they need to be analysed, prioritised and a more thorough understanding of the new product concept needs to be formed. This allows the view of the entire release to be developed, thus the added value for the buying companies is formed. This work can be enhanced with the use of prototypes and / or customer comments.

Next, the selected new release specification is accepted. Depending on the implementation of the model used, some of the specifications can be decided until the implementation phase, or the specification is frozen at this point.

When multitasking, after the project decision the marketing can start partial work in parallel with design and implementation. This may be difficult due to the fact that at this point they have little to work on. The documentation and people responsible for the testing of the product should be brought in at this stage. When they understand the actual design and specification, they can start the documentation and planning the testing already at this phase, not after the product is fully implemented. The "strictness" of the order of these tasks is naturally contingent to the environment (internal and external) of the company.

After the product is finished in terms of implementation, testing and documentation the sales and marketing can start working without any of those issues hindering their work. As mentioned earlier, some of the sales and marketing work may have started earlier based on specifications / prototypes. When the documentation has reached the first good quality version, which can be shown to the customers, the expectancy of concrete sales

results increases significantly. After the release is out in the market, it is a good time to gather feedback to be used in the future release again.

It is noteworthy that the interaction mechanism between the new product development unit and the marketing unit is essential for the product company to excel in producing market- and customer-oriented deliverables. Ottum and Moore (Ottum & Moore 1997, 260) note regarding their study of 58 technical products in different U.S. companies that: "However, just gathering information is not enough; sharing market information across functional areas is also critical to success. Furthermore, it has been observed that sharing of market information is problematic for many firms. In general, market information is perceived to be "owned" by the marketing department and is rarely shared with other functions. [...] During less-successful new product development projects, market information moves out of the marketing department only in infrequent batches. On the other hand, developers of successful new products transfer more information of all types across the functional interfaces in a continuous manner."

One interesting point is the emphasis on documentation, as according to (Cusumano & Yoffie 1999, 60-69) most (PC and Internet –domain) companies emphasize code production and are willing to tolerate incomplete documentation, which can be seen as a rather common problem in the software company world. This type of situation also suggests a common possibility for improvement and still a rather technically oriented internal development process, instead of a market or sales oriented one. Especially in the Business to Business (B to B) MOTS software product domain it can be argued that if a software company's processes value the generation of code over documentation and simultaneously it is common knowledge that it is virtually impossible to reach sales with the important, large corporate customers without extensive, good quality documentation, this suggests that the process of the software company is not as market driven as it could be. Instead, it is then a more technically oriented process.

4.4 Requirements management

This area is under almost constant evolution during a young product company's development. In the beginning it is common that this task is done by 1-3 key people on

the side of their other tasks. Compare to FIGURE 3, where there is a small core development team responsible for requirements while they are also implementing the features. As the software company matures the separate role of product management is established to handle this task more efficiently. A traditional, tailoring-oriented approach is very different from the experienced situation in the product field, as tailoring company does not have to constantly develop any software item according to unspecified or partially unclear market needs. While this situation is reached, the organization and / or product are often growing to the extent that is no longer possible to handle all the feature suggestions, architecture modifications and other requirements only by word of mouth. If not done earlier, acquiring commercial requirements management (RM) software to offer formality and to enable the communication between different employees can be beneficial. This type of computer-supported collaborative work (CSCW) is not only beneficial to the developers and the NPD unit of the software company. At best this type of system can serve as a mechanism for the different units (NPD, Customer Service, Sales and Marketing) to be able to input and share information of the requirements they have noticed. If the system supports giving comments and evaluations of suggested features, this can be even more beneficial. Without any such mechanism the software company may be locked into a situation, where it can benefit only poorly from its increased size, as there may still be only few people who can actually contribute to the new features of the software. This type of situation does not support the organizational learning from the customer and product domains optimally.

Käkölä and Salo (Käkölä & Salo 2002, 14) refer to the following table of phases related to requirements management. This allows a very procedural view of the whole activity in question.

TABLE 4 Phases of requirements management

Phase	Description
Capture	The process of collecting new product ideas and requirements from all relevant sources and with minimal control on representation or content.
Categorization	The association of submitted requirements with appropriate context.
Refinement	The transformation of requirements information into units about which a tentative product decision can be made.
Assessment	The evaluation of requirements information by all relevant stakeholders in view of their alignment with the criteria and targets that have been placed.
Follow-up	Systematic monitoring of requirements usage during the later phases of the product development process.

As the phases are in logical order, there is also another level of thought to this issue. The following presents a comparison of the high-level process in software tailoring and in software product development. A software development model is presented by (Brownsfold, Oberndorf & Sledge 2000, 49) (adapted).
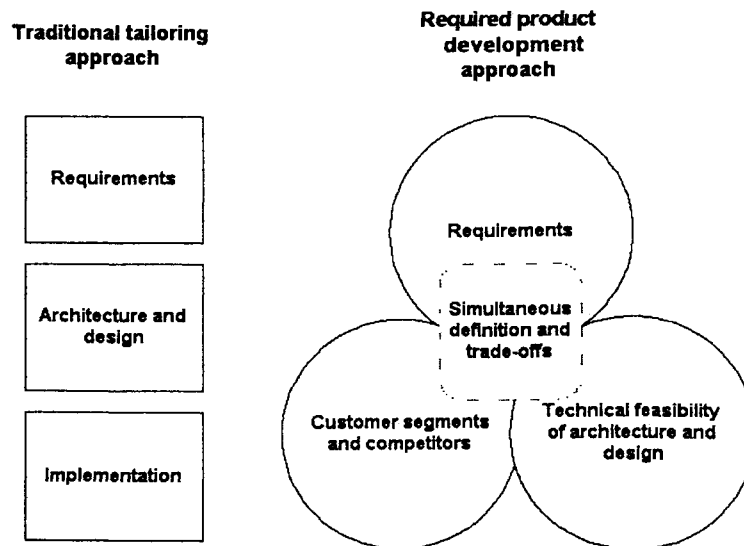
FIGURE 7 Comparing two different RM and software development approaches

The left side of the figure presents a portion of a traditional tailoring process. In the product domain there are more complex interrelationships between affecting factors. As the requirements gathered (possibly with the help of some type of CSCW-system) are listed, understood and categorized, there are still higher-level issues to solve. Selecting the customer segment / customers targeted with the next particular release of a software has a bearing on the selected features, as do the competitors' actions also. Especially in the case of a young MOTS product company there may be only a small number of targeted customers, which have significant influence on the future releases. In this type of a situation the MOTS product company's skills in relationship marketing are valuable. As the next release is often expected by the market to be in the sales phase in a certain time, this puts pressure on the NPD of what exactly can be accomplished in a *limited time*. There also the issue of what can be technically accomplished with a product in a *commercially feasible fashion*. The product revenue seldom increases by manifold due to high costs of building a single feature or better internal architecture of a software product, even if those would be nice to have.

The strategic importance and difficulty of crafting the right architectural decisions and correct feature set is increased in a situation where the product market in question is an undeveloped one and the customers are not necessarily able to identify the exact set of needs they have. As the market matures receiving reasonably homogeneous information from the customers and partners is more probable, but even then the task of crafting a commercially optimal set of features in optimal expected development time for a targeted segment of customers can be a complex task.

After the perceived optimal set of features has been reached through some internal decision-making process, that definition can be passed on to a project manager and his development team. If the perceived development timetables estimated by the software designers, project manager and other related staff have been correct, and the market vision crafted by the product management is correct, there is a good possibility that the product will arrive to the market in planned time, which is competitively beneficial to the product and the company. This issue of the product company's approach to the market is discussed next.

4.5   Approaching the market

A discussion has been presented in earlier chapters regarding the software product company's internal processes and the internal growth to organizational maturity. This discussion has progressed to a point where rather mature internal processes have been described. In this and the following chapter distribution and logistics strategy, and product strategy for a software company are examined in more detail. The matters of pricing strategy and communication strategy are also briefly discussed.

When defining the business and licensing models of a software company one must consider several factors, which may influence the performance of the whole company. A good combination of a business model and licensing can at best create a prosperous mechanism of the company interacting with the selected market segment. In the worst case the mechanism may seal the employees of the software company to a poorly profitable working model.

Regarding the distribution and logistics strategy McHugh (McHugh 1999) suggests that there are two main factors contributing to the choice of the business model. These are the entry-level deal size and the average length of a sales cycle. The sales cycle is typically dependent on the complexity and price of the software. Clearly, low price, low complexity and short sales cycle enable the software company to use low cost sales channels (Dealers). When the price or complexity increases, also the sales cycle tends to grow longer. This in turn suggests more expensive sales channel to be used, such as the VAR or OEM channels. A greater deal size can justify other partners to contribute significantly to the implementation and installation, as a smaller deal size suggests the software / service to be sold with less tailoring work done by other parties.

Parvinen (Parvinen 2001) lists the following sales channel selection (adapted):

- direct sales

- agency

- joint sales with partner

- OEM (Original equipment manufacturer)

- professional distribution organizations

- VAR (Value added reseller)

- agents and retails

- mail-order selling

- internet as a direct sales channel

- internet as a supportive mechanism for the sales channel

The critical issue here for the software company's management is to understand which of these options serves best the future development of the software company. It is not only the issue of calculating how much the customer will pay and how the X amount of revenue is then divided between the parties involved. A deep understanding of the sales cycles, tailoring resources needed per single customer and the annual throughput capacity of sold software licence sales is required. This should give the management a reasonable overall picture to compare with the costs of running the software company.

Especially in the case of a new software product the software company needs an interaction mechanism with the first customers in order to learn how the product should be improved in the future releases to aim to the whole product status in the eyes of the first customer segment. This seems to be often down prioritized in the partner / channel negotiations of small software companies. A negotiation result can be considered good when a big-name company has promised to start selling the smaller company's products, without any clear interaction mechanism for any future improvements of the product. This may not be enough in the long run for the small software company. Especially the OEM channel poses this potential threat to the software company – the sales channel may want to own the customers "too much", without letting sufficient interaction to happen between the customer and the software company. In a case of COTS sales this may not be a problem. But in a case of MOTS sales where the software product company may only have a few paying customers, gaining the feedback and having the sufficient interaction with the customers grows very important.

One potential factor for this type of possible problems for MOTS companies also can be that the sales-unit may often be primarily interested in short-term sales – e.g. they may

not be very eager to take into negotiations the interaction issues regarding future development of the product. This can be seen more as a task of a research and development unit, not the sales unit, who is nevertheless carrying the negotiations with the potential sales channel. In this type of situations the role and presence of the senior management grows very important. The clear requirement here is that there is someone within the software company who is able to look the situation holistically and determine which negotiation tasks are prioritised high. Another way of preventing this type of situation from occurring is the use of cross-functional teams within the software company. This type of approach is used widely within Hewlett-Packard (HP). This type of management team can have managers from different units of the software company (Customer Service, NPD, Sales, Marketing) preventing the optimisation of any single unit's goals and making sure the company's requirements for the interaction are accounted for holistically. This team can then be responsible for the interaction with the customer. This information regarding different sales and negotiation mechanisms is based on author's personal experience in software companies and may be referenced as such.

4.6   The suggested product sales and marketing development according to Geoffrey Moore

In the previous chapters the managerial planning of the company's market approach has been given a large emphasis. After planning it is time to start implementing both the products and the marketing plans. In this chapter a sequence of software company and product's movement to the market is described and adopted from (Moore 1998), (Moore 1995). This can be characterized as a strategy framework for emergent MOTS products and companies. However, the MOTS-context is not defined to be the only area where this framework is applicable. This selected framework covers the period from the beginning of a development of a product to the point where the product has already had commercial success and is ready to be targeted to mainstream customers of a market. This development in the technology market is described as follows:
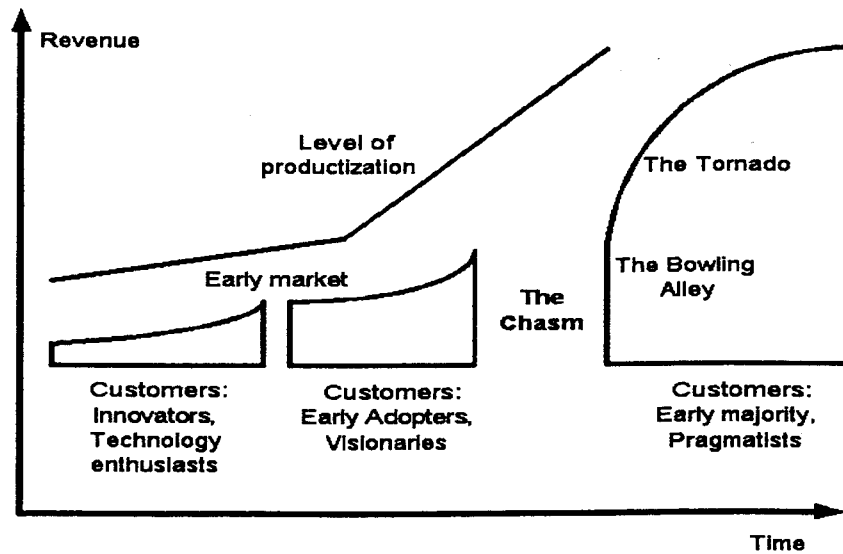
FIGURE 8 Suggested software product strategy

According to the suggested model the first customers (company buyers) are **innovators and technology enthusiasts** (techies) within a buyer company. They are characterized as "those who appreciate the technology for its own sake", or as "Gyro Gearloose (Pelle Peloton in Finnish). They do not require a high level of productization; instead they are more interested in the architecture and technical features and benefits of the product. Companies with such type of enthusiasts are typical customers of software products in their early development stages.

The second group of customers is the **early adopters, visionaries.** They are characterized as "people who have the insight to match an emerging technology to a strategic opportunity." This group is still not requiring very highly productized solutions, although the requirement level is clearly higher than in the first group. The risk in this group is that they may want things that are not consistent with the product concept sellable to other customers.

The third group, the **early majority, pragmatists** are people within the buyer company who do not want to be pioneers in the technology adaptation. They are characterized as

more like the X-files' Scully than Mulder, or more like Lethal Weapon's (Tappava ase in Finnish) sergeant Murtaugh than Martin Riggs (Mel Gibson). This group requires a very high level of productization and proof in the form of reference list how the product has been performing in other installations. This group does not welcome risk in similar levels as the former groups.

The following discusses how can these groups be used to benefit the software company and how can they be utilized through some cognition.

The first phase is naturally to find *some* customer(s) for the new product, in order to really focus the product development and acquire knowledge regarding the field. It seems credible that considering the poor quality of a typical "beta-level" release 1.0, the innovators and technology enthusiasts are among the very few, who are willing to accept and pay for such a "product."

In the second phase visionary customers are required to further develop the feature set and level of productization of the product. This is necessary for the software development unit to be able to create a productized release to meet the future customers' requirements and expectations.

The Chasm presents a time period, where the product already has gained customers, but the company has not yet been able to establish themselves as the leading supplier in any customer segment. In this situation the company is forced to act as the defendant against the leading suppliers of different customer segments, hence reducing their capability to utilize premium pricing for their product, thus weakening the software company's situation in the long term.

At this moment the software company has several different segments of customers it has been dealing with. Each of these segments has their own requirements for the next software release(s). In the context of overlooked segment selection Moore defines the sales driven company's stand (as opposed to market driven company) as follows: "*We do not have, nor are we willing to adopt, any discipline that would ever require us to stop pursuing any sale at any time for any reason.*"

Unfortunately no software company has resources to fully satisfy everyone. Nevertheless the management of the (young) software product company typically does not want to put down any of these potential segments, but instead pursues all the possible deals in sight. What is done, the requirements are collected from different customers / segments and version requirements definition is crafted by the product management containing the features seen most important for different segments. The software release is implemented which offers something for every segment. In Moore's description "it has something for everybody", but the downside is "it has everything for nobody." According to Moore, the pragmatists only buy whole products and are not ready to invest their own resources to develop the product to a required level. This decision mechanism suggests that the company's stay in the chasm will be prolonged until it has been able to focus on certain segment's needs and has been able to implement (codify) the knowledge and requirements gathered from that particular segment to its own software product. This is the behaviour of a market driven company. Until this is done, the software company will not see success in the commercially most important pragmatist group.

In the third phase (regarded as the bowling alley with bowling pins) the aim is to present and further focus the current product and its potential to a single customer segment (single segment's metaphor is a bowling pin). At this time the software company is concentrating on representing a productized solution which is compelling enough for this particular segment to be adapted by the companies of this segment one after another. For this to happen the software solution should have matured to a level of a whole product. In the fourth phase this segment is then used as a credible reference for the next pragmatist group.

Fourth phase includes either targeting another segment (bowling pin) with a product modified for their needs or reaching directly to the goal of targeting the product for the large, mainstream market, characterized as the tornado. (Rajala et al. 2001, 52) have analysed Moore's work and drawn a table of activities in each phase. This table is adopted from them.

TABLE 5 Activities in different phases of product development (Rajala et al. 2001,52)

	Innovators, Technology enthusiasts	Early adopters, Visionaries	Early majority, Pragmatists
Product development	Core product development	Product-related services & features	Fully integrated, commoditized whole product
Servicing & Implementation	Lot of custom service	Product related services	Whole product services
Marketing & Sales	Direct (Partners can assist)	Indirect (Sales partners)	Various alternatives

In Marketing and Sales one factor (not mentioned in the original table) for the added value of sales partners is the complexity of the product. Especially in the case of a new product from a new product company, it is not presumable that the sales partner will invest and work with the product to the extent that they are able to answer all the upcoming issues with the customer(s). This means that in the case of a complex product, the role of the software vendor is to assist / participate with the sales and installation, at least with some customer cases, if not with all of them. This requirement for slightly different cooperation process can be seen as an implicit requirement in the MOTS area due to complexity caused by tailoring and the relevant negotiation process.

What is noteworthy in Moore's writing is that "*the pragmatists evaluate and buy whole products.*" At the same time they often prefer to buy market leaders' products. Furthermore, it is suggested that a large portion of managers who are responsible of companies' technology infrastructure purchases can be positioned to this buyer segment. Since this is also evaluated to be the largest customer group in terms of revenue from the product and related services, it seems significant to understand this situation regarding any new product's potential success.

It is notable that Moore's work originally includes the market positioning even further than what is described in this study. After the success in the market Moore introduces such customer groups as the Late Majority and the Laggards, and the descriptions of their different behaviour. However, since the object of this study is "only" to examine the software company's journey to market success, and not beyond, these later phases of Moore's original model are not examined further in this study.

The importance of the presented model is to describe what kind of customers the software company should seek / expect in a given phase and to be able to answer to the different behaviours of these different buyer groups. The understanding of this is essential and can be used as a tool to focus the software company's efforts in a given time of the company's life. While the development of internal processes is equally important, the company's success in the international software market will finally be measured by its external market activities. Failing in this area jeopardises the whole company and the internal maturity can do very little to remedy the situation.

This strategy framework characterized the different groups of customers and their relationship to the software product in different phases of the product development. In the next chapter the MOTS product business area is examined in more detail.

4.7 Competencies developed within a product company

While the starting point of this thesis is the tailoring company, it is notable that the competencies listed here are largely new requirements for the tailoring company. The contents of this chapter are mostly a synthesis of the issues discussed in chapter 4 with its subchapters and serve mostly as additional requirements to the competencies developed within the tailoring company. Assuming that the software company's management has already done all the strategic and planning work related to the selected product domain, there are still clear, practical areas which are lacking planning and attention. These tasks may not be accomplished adequately if they are left for the development / sales staff to design and implement. Product oriented functions of a software product company include the following (please note that the list is not meant to be exhaustive):

- Management skills in planning the different strategy areas and their implementation.

- Product and feature management and innovation, as this area can be concluded to be one of the most important competitive areas - processes, talent and competencies are required.

- Market insight, the software company and its success may be dependent on the success of the product, the understanding how the product-specific market is developing is essential.

- Requirements management, when the product and the company grow larger in size and more mature understanding what features are essential / should be invented, is a major factor for the company's success – this area is related with the market insight.

- Marketing, product marketing in an international environment is something very new to often domestic tailoring companies.

- Version management, as it is typical that MOTS customers require additional, customer-specific features, controlling the versions and their customer maintenance is challenging (and often not within the experience of the tailoring companies).

- Documentation, the requirement level for software product documentation with documented integration points, scalability, multiple different setups, multilanguage support etc. is a large task requiring dedicated personnel.

- Extensive product testing, which cannot be carried out solely on the customer's premises.

This chapter has discussed the qualities of a software product company. As some of the MOTS-specific qualities have been mentioned when it has been contingent to the contents of the examined issues, there are still some MOTS-domain specific issues, which have not been treated. Those issues are discussed in the following chapter.

5 OVERVIEW OF THE MODIFIED-OFF-THE-SHELF SOFTWARE BUSINESS

This chapter examines the factors of the MOTS software domain, which are not common to other defined types (COTS, Tailored). As is described in the following chapter 5.1 MOTS Business model on page 60, this domain is mostly B to B trade. Hoch et al. (Hoch et al. 1999, 33,36) categorize enterprise software solutions and give examples such as decision support software, call center software and relationship management software. According to them this category always needs customisation and typically has installation periods of several months. The qualities of this group are very similar to the MOTS software product definition. Though the installation time can vary greatly depending on the nature of the implementation. The main indication here is that the cycle from first contact of a customer to finished installation can easily be 12 months in the case of small companies' product – this is something to take into consideration when crafting company's financing and business models.

A question can be raised why a software item is called a product, if it still needs tailoring and the delivery time can be several months? Let us have an imaginary example: The goal is set within an enterprise to have a new software service. In this customer segment there may be 1000 similar customers with potentially the same motivation. Two facts can be concluded here:

1. It is not feasible doing 1000 projects from scratch, thus this does not justify the usage of the tailoring-focused approach.

2. Each customer may have a different software and / or hardware environment of their own and require the new service to interconnect to their existing system(s). Other reasons may be that they want the service to be customized to their look and feel, or they simply want to add n new features before they adapt the service. This doesn't justify the usage of the Commercial Off the Shelf (COTS) products either.

The MOTS approach is to create a software product, which can serve as the core product for the product installation. This core product may have a high amount of parameters, which can be set at installation time to meet the single installation's purposes. Everything that cannot be taken into account in the earlier product development phase is done as tailoring / customisation work for the customer.

In a more complex example the software company may have several such products, which can be combined at installation phase to create the required functionality with some tailoring work. The beneficial side of this approach is that compared to a project, the delivery time can be reduced greatly and the overall quality of the end-result can be significantly higher, since the software solution is used and tested also by many other installations and users.

Another quality of MOTS business is the sales of licences. In this area an interesting combination of COTS and Tailoring practices occurs. According to (Warsta 2001, 163) The MOTS type of delivery includes a system licence contract that gives the customer a right to use the software. However, the actual sales process with the customer is again a systems sales process – i.e. tailored software sales process – during which the supplier defines the customer's needs and the interface between the supplier's proprietary software and the customer's environment. Included in the systems sales, they (software company) describe the solution of how to solve the customer's problem and its costs and the software components it includes as well as the delivery dates and places.

What is noteworthy in Warsta's statement, is that the MOTS sales process is really not a product-oriented sales process. This is an important point to understand for the management of a software company. There is a pitfall that the expectations of revenue and sales can be high in the young product's / company's upper management, since the company is in the product business. Sales and revenue can be expected at the speed of COTS product sales, after all the company is selling software products? However, this is clearly not the case in MOTS business. The negotiating + analysis + definition of the requirements alone can take months. It is only after, when the implementation, installation and testing can start.

What Warsta is not describing in this relationship is that even when the sales process is similar to tailored sales process, the focus of the sales and marketing communication is not. When a customer is presented a tailoring company's presentation, the arguments and contents are about reliability and performance of the *company* almost regardless of the project contents. On the contrary, in MOTS product business, the contents of the presentation are focused more on the *product's* qualities and performance, naturally giving some emphasis on the reliability of the vendor also. This change in emphasis causes differences in the negotiating process – one clear difference being the requirement to show at least partially the solution prior to closing the sales in product business.

Another point of interest is the testing of the installation. In the tailoring business it is natural and commonly accepted that the tailored solution goes through a rigorous testing when it is turned over to the customer. On the other hand, in the product business it is often an implicit assumption, that the *product* is tested after it is developed and there is very little need to extensively re-test the software in customer's premises during every installation. However, in the case of MOTS products one cannot really say that the whole system is tested thoroughly in the product development phase if there are new software functionalities added to the system and old ones modified during the tailoring phase. This suggests that in the case of MOTS business, even when discussing of products, the testing of the installation should be carried out similarly to tailoring projects, instead of COTS products. If this suggestion is accepted the revised testing task should also be clearly reflected in the process description of the Customer Service-unit (CS) responsible for the deployment of the software product. This requires resources, additional time and a slightly different communication process with the customer, when targeting a mature overall process of delivery and deployment. This may sound all too natural when discussing of ERP implementations, nevertheless it may not always be the rule of thumb in the smaller MOTS product company's CS's process when discussing of smaller-scale product installations.

5.1   MOTS Business model

Let us take a further look at the MOTS focused company and its business model. First of all the company is selling licences and related tailoring work instead of just labour, as does the tailoring company in its Business to Business relationship. In comparison, the clear goal of a COTS focused company is to develop and sell software, which can be sold for as many copies as possible. Each copy typically includes a licence to use the software. Typically this type of software is sold from B to C (Business to Consumer) due to the low licence prices affordable to consumers. When the COTS software is sold to companies (B to B), its qualities typically do not change significantly. Since there is virtually no extra work related to selling yet another additional copy, there is very little costs related having this copy to be installed by the end user himself.

The MOTS focus differs in approach somewhat. Naturally there is the same motivation to sell licenses as is in the case of COTS business. However, since there is always tailoring work required, the amount of sold licences increases the workload of the company almost linearly. As just the tailoring costs of MOTS products are always very high compared to the total costs of COTS products the trade of MOTS products is always from B to B. The consumer market simply does not support this type of cost level for any known software-based service. This cost level combined with the natural complexity of the deployment (with the tailoring) also has its effect on the choice of the sales channels, as not all the sales channels are resourced to handle such tasks (such as the dealer channel). The following FIGURE 9 describes the business model of the MOTS company.

```
┌─────────────────────────────────┐
│   Lisensor with Platform / Core  │
│             product              │
├─────────────────────────────────┤
│  Tailoring task specified through│
│           negotiations           │
└─────────────────────────────────┘
                 ↕
┌─────────────────────────────────┐
│   Dealer / VAR / OEM -channel    │
│  possibly involved with tailoring│
└─────────────────────────────────┘
                 ↕
┌─────────────────────────────────┐
│      Customer and Licensee       │
│                                  │
└─────────────────────────────────┘
```
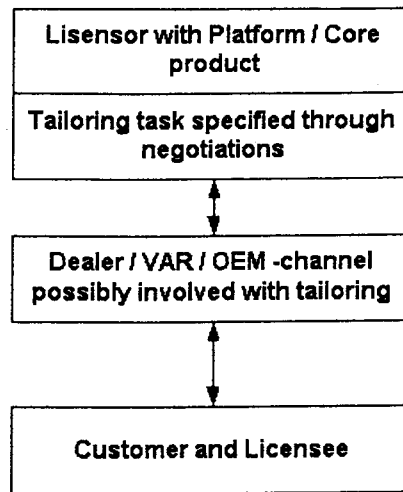
FIGURE 9 MOTS licence-based business model

Since licensing offers a very abstract way of passing a defined set of rights to another party, this can lead to very complex licensing models. This can be seen as a contractual issue, but is also very relevant to the selected business model of the software company, thus it is relevant to the whole company.

In a very complex example a software company may be selling licences to a second party, who in turn combines or bundles this licence with a combination of other licences from other companies and sells the whole bundle as a licensed combination to the actual company utilizing all these software services. A second more complex example is a situation, where the software company distributes clients / development environments for free, but then in turn tries to price the server licences for the clients / runtime-licences for the software created with the development environment. A sales channel can be in the middle of both of these examples.

Depending on the sales channel used, the channel may carry out some tasks of tailoring, installations and support after the installation is finished. The channel is bound to have a bigger role when the cooperation mode changes from dealer (typically very small role in tailoring and support) to Value Added Reseller (VAR) to Original Equipment Manufacturer (OEM) mode. In OEM mode the original software product company is

supplying the software to the sales channel, which is motivated to offer as much tailoring services related to licence sale as possible. This allows the OEM channel to own the customer relationship in its entirety. This contributes to the OEM channel to be able to own also the support revenue to a large degree.

Most common modes of cooperation for a small MOTS company are by nature VAR and OEM. If the product has emerged only recently, this can contribute to the tailoring requirements being still at a high level, which in turn suggests that the dealer channel may not be able to handle it. Being without any domain-specific expertise and allocated extra resources, the dealer channel requires well-productized solutions without any large learning tasks included. In the case of small software company and / or a new software product the motivation for a sales channel to invest heavily on learning the new product may also be low. This can be seen contributing to a direction where a majority of new MOTS products may be using sales channels, but a large portion of the tailoring work is done by the original software product companies. In a more mature situation (such as the ERP –market, where there already is an established market) the single sale can be so substantial and the market so well developed that a good business motivation exists for the sales channel to learn to do also the tailoring work required for the product deployment.

5.2   MOTS tailoring and interaction with the customer

The company customers buying the MOTS software products typically want the purchased software product to interact with their existing systems. The technological environment in different customer companies is very often heterogeneous. More often than not, the new MOTS software needs to interact with more than one of the existing software components (and each of these existing components has n different implementations on the market). When selling their software product, the software company needs to make a proactive, credible statement how they exactly plan to assess the situation. In this type of setting it may not make sense to try to prepare the software product for all these possibilities. One feasible tactic is to select to most common

components to support readily within the product and offer additional support on a case by case-basis.

The assumption that the integration work between different, formerly undefined systems could be achieved only through parameterisation and productization seems very challenging (this is the kind of work, which makes companies like TietoEnator in the Nordic countries prosper). This is true especially in the case where the negotiating power of the first customers with their requirements is quite substantial.

Tailoring has also another significance. This is the process when the final requirements from the customer are typically received. This is also the process when much can be learned *about* the customer(s) and customers see this as the learning period about the software company. This puts emphasis on the interaction between the parties at this stage. If the software company is able to present a credible plan and execution of the implementation, installation and testing, it has a very good position to sell another product release in the future. In turn, if the company is performing below expected level, missing deadlines, not seeming to be in charge, or even understanding all the related issues, the first installation may pass adequately, but the next installation / upgrade with the same MOTS customer may be far away. Establishing good, credible references after the tailoring process is essential for a small product company. In this type of a situation both the project management skills and the product business competencies can and should be used jointly.

Athaide et al. (Athaide, Meyers & Wilemon 1996, 406) examine the relationship marketing activities employed by successful sellers of high-tech process innovations. They identify eight strategic marketing objectives that underlie these relationship marketing activities: product customisation, information gathering on product performance, product education and training, ongoing product support, proactive political environment, product demonstration and trial, real-time problem-solving assistance, and clarification of the product's relative advantage. Their findings suggest that successful sellers engage in relationship marketing activities throughout all phases of the commercialisation process. Rather than simply trying to close a deal, these firms seek active involvement from potential customers, ranging from codesigning of

products to seeking feedback on product-related problems or desired modifications (Ibid).

Another indication towards the close interaction between the software company and its customers is offered by Souder et al. in their study of 29 New Zealand-based small, entrepreneurial high technology firms (Souder, Buisson & Garrett 1997, 470). According to their research results building strong interfaces between NPD personnel and counterparts in customer organizations is important. Their studied N.Z. firms were all encouraging their personnel to spend long periods of time at customer sites, both domestic and foreign. "According to the managing director at one N.Z. firm, "We need to know customers well enough to accurately anticipate their needs five years from now." This type of approach to customer intimacy can be seen as applicable to the MOTS domain as well. The approach described here suggests that it may be difficult for the software company to gain knowledge of the domain, if the interaction with the customers is carried out only by partners.

When the MOTS software product is offered to the customer company, the issue of customer intimacy is often realized in a form of a software trial period. As there is often tailoring required for the product to interact with the customer's existing systems, this cannot be carried out similarly to the COTS products. When the negotiating power of the customer is large (this is very typical for small, starting MOTS companies), the software company may not be able to charge a large amount of money for the trial, not to mention any additional charging for tailoring for the trial purposes. In this situation a feasible approach can be to identify what exact components of the software the customer is willing to accept based on demonstrations and what exact features seem to be significant for the trial. Negotiating the trial to include only the parts seen mandatory by the customer can reduce the non-chargeable work required to arrange the trial. This can act as a contingency factor for the trial to be accomplished in a relatively short time frame and the customer to focus on the relevant, agreed issues. In a more immature situation the sales people are motivated to promise the whole product to be trialed in a very sales driven manner. Failing to do the necessary trial planning will result to prolonged installation and trial periods with decreased profitability of the product sale. This aspect is again very different from COTS and tailoring domains.

The tailoring company and the product company have now been described, and finally the implications of MOTS business have been examined. What is left of the first, theoretical part of this thesis, is to examine the motivations for switching from one domain to another.

6 THE MOTIVATION FOR MOVING FROM TAILORING BUSINESS TO SOFTWARE PRODUCT BUSINESS

In the chapter following this one a company case study will be presented. Before the case study it is essential that the motivation for a company to execute such a transition from one business mode to another is understood and analyzed. One can name several examples of motivation in the company management level (this list is not intended to be exhaustive and is based on studied company X material (Appendix 1,2) and the work of Roberts) (Roberts 1990, 275-287):

- When the company has participated in a certain field for a longer time focusing on technical and project perspectives, it is easy to feel that the technical part for the product domain is already accomplished at large part.

- From a project-experienced person's point of view the idea of receiving licence payments from the same software repeatedly with almost zero reproduction cost seems lucrative.

- Since there is typically little strategy / marketing knowledge regarding MOTS software products in the tailoring company, the task of successful market entry is often confused with technical issues (which are thought to be in control), such as having technology X within competence or having a component of 100 Java-classes finished. At the same time the company strategy and marketing tasks are evaluated as something that can "almost" be accomplished with current resources.

- It is very easy to try, since the tailoring / project business is thought to function "on the side", while the product business really starts up.

- Company's management / owners have adopted a growth strategy which needs to be implemented (Appendix 1).

Hence, the most positive viewpoint can include clear revenue growth (even with less work after the first releases are out), utilizing the existing technical implementations and

competence the whole venture can be accomplished with minimal risk, since it is not necessary to shut down the tailoring business at this stage.

Considering this background the motivation for the transition seems rather self-evident. Here is a SWOT-analysis of the situation in the beginning.

TABLE 6 SWOT-analysis of the tailoring company targeting MOTS product focus

STRENGHTS	WEAKNESSES
- Technical experience - Existing implementations - Existing personnel	- Little product marketing experience - Not all the required competence areas are covered within the company - No clear understanding of product strategies
OPPORTUNITIES	THREATS
- Revenue potential - Possibility to increase the company's net value - Career opportunities for the personnel involved	- The finalized product may take far more resources and time than anticipated - When finalized the product may not be competitive - Lack of management / marketing competence and resources may miscarry even a competitive product

These type of circumstances can serve as a motivation for the decision to go forward with the approach to the MOTS software product business. In the beginning this is resourced low (Roberts 1990, 280-287) and personnel may be advised to participate in the effort when requested to do so. Some key personnel may be dedicated to promote the product approach. Often (some of) the persons carrying most of the responsibility for promoting the product are newly hired to work with this task (as in the case study of company X in the next chapter).

In his study Roberts (Roberts 1990, 274) states: "Research studies of 114 technology-based firms within the Greater Boston area indicate evolution over the first several years

after founding toward more product-oriented businesses and away from consulting and R&D contracting, and increased orientation of the founders to sales and marketing, with lessened emphasis on engineering." Furthermore he notes (Roberts 1990, 279) that some products were born through indirect or unintended cause of action. "In other instances the development of a product was the result of a conscious decision to change the character of a firm. This was especially true for several firms that had initially been engaged solely in contractual development work. Although this work often provided a stable source of entrepreneurial income, a number of the founders of these firms realized that far greater profit margins and better opportunities for corporate growth existed in the sale of products. These companies set about deliberately to develop products..."

The relatively early point of this very similar study approach of tailoring companies focusing on the product domain indicates this type of transition from one business mode to another to be a relatively old, but little examined phenomenon. While this is the conclusion for the motivation for the company's transformation, the next chapter examines a case study of a company X attempting such a task.

7 CASE STUDY

In this chapter the longitudinal, reconstructive case study research is described. The progression of company X's two selected MOTS products during the two-year period of 2000 – 2001 are studied. To give context to the whole research case X's overall environment is shortly described first. During the course of this research, appendices are used only to verify the findings already stated in the following chapters. It is not mandatory for the reader to acquaint himself with the contents of the appendices in order to comprehend the contents of this study. Furthermore, to preserve anonymity of the actors of the case, all identity information has been removed and the appendices have been determined as classified information.

7.1   Background of the case company X

The author of this thesis worked in two managerial positions during 2000 – 2001 in the company X offering mobile applications to the teleoperator segment globally. Other identified customer segments were portal owners and service providers (ASP, application service provider and ISP, internet service provider segments). This is show in (Appendix 1,2). It is noteworthy that the only realized cooperation regarding X's product area had occurred in the teleoperator segment by the Q1/2000.

This Finnish company had started software-tailoring business some years earlier, thus in 1999 it was still a rather young company of some 70 employees. The organization was divided into two tailoring units of which the other one was named as the mobile unit (Appendix 2). The mobile unit had some 30 employees in 1999. Before year 2000, X (mobile unit) had done some tailoring projects to a domestic teleoperator, thus it had experience in the technological area. The prior projects involved SMS-messages (thus the SMS-C was used as a connectivity gateway), WAP-interface (similarly the WAP-GW was used as a connectivity gateway) and WEB-interface (a web server software was needed in addition to an application server software running the Java servlets – this was before these components were more commonly offered in a single application server package) to be used in connection of a single server application. Normally a

single installation had more than one of these interfaces installed simultaneously and the end-users (customers) of the teleoperators could choose their preferred interfaces to be used. This can be described as having multi-interface server applications.

The applications were developed using Java (J2EE, servlets) as the development language, and relational databases (such as Oracle 8.0.x standard) as storage. The whole solution was rather easily portable to any common operating system having a Java Virtual Machine (JVM) available for that particular OS. This design was to give freedom to the customer to choose the hardware environment to be the very same they were already familiar with. Another advantage of good portability can be seen the easy partnering with different hardware vendors, since in this case it was not a large task to support a hardware vendor who in a particular case had the best relationship with the customer in question.

Given this organizational and technological background the company's management made the following decisions during Q4/1999 – Q1/2000: At first a public strategic decision was made that a growth strategy should be followed. The second public decision was that the mobile unit should be responsible for the international product business targeting for growth. Third decision was that there should not be any excessive risks involved, so the mobile unit would continue carrying out tailoring business (creating revenue) while some of the unit's resources would be allocated to developing the results of earlier projects into more productized form. These issues are noted in (Appendix 2). According to (Hoch et al. 1999, 37-38) due to the differences in cost structure, demand volume, competition intensity, geographic presence and relationship management the two software business types (tailored and product) are so different that they are very difficult to manage simultaneously, also compare to TABLE 1 Software product business compared to tailored professional services on page 21. As X did not hire any internationally experienced senior management, this (and other(s)) decision was made based on the management's experiences from domestic markets, which was mostly from tailoring area during the very few past years.

Furthermore, the upper management of X communicated publicly the strategy of the company as developing products and services to the teleoperator and other segments globally, as is again noted in (Appendix 2). The strategy did not explicitly indicate what

exact products or services were in question. Instead of market focused approach the upper management had a very sales oriented approach. All the possible markets and products were included into the focus areas. The products were targeted to the teleoperators, application service providers and internet service providers. Any additional segment, which could utilize the applications, was also considered as a possible target area. According to the same sales oriented approach, the small sales and marketing team (less than 5 people) should not limit themselves to any nearby area (such as Europe), instead the whole world was defined to be the focus area, since sales could come from any continent. The sales and marketing team consisted of young personnel who had little or no experience of successful international product sales. According to the sales driven approach used, the exact product focus of X would be more clearly decided when a good opportunity arises. The plan was that more resources would be allocated to product business when it really starts up.

Regarding the marketing approach, a relatively heavy emphasis was put on partners acting as sales channels, as most of the sales and marketing staff were allocated to assist the sales channels. Three large corporations were negotiated to act as sales channels for the products. In the beginning of the cooperation X had expectations that these channels would be able to function in an autonomic manner and bring in clear orders and strong revenue. The situation quickly started changing regarding the channel management when it became evident that the large corporations were not interested in spending resources to learn the server-side products thoroughly, but instead always needed strong assistance in the functionality and technical aspects of the products. Especially the tailoring part of MOTS products was clearly outside of the competence of the partners. The partners were clearly unable to negotiate what could be done to the product, in what time and what should be charged for that work. This seems rather understandable given the fact that there was no documentation on the internal architecture of the products to begin with. This led to a situation where the emphasis on the sales channels remained, but the resources of the sales and marketing unit were more heavily allocated on the support of the partners acting as sales channels, thus reducing the direct sale's resources to minimum.

7.2   The MOTS nature of the products

A certain part of tailoring is always included to the nature of MOTS products. When designing a multichannel entertainment service to a customer the following tasks had to be accomplished (among others):

- The product had to be localized which required manual translations and their installation and testing with the software.

- The product was always requested to be customized to the teleoperator's look and feel (all interfaces).

- The product had to be interconnected to the existing SMS-C, WAP-GW and WEB-services. All of these gateways are heterogeneous in technical implementation, which lead to the work being done case by case.

- It was learned in the later stages of the market entry that the different character sets also presented yet another issue regarding localization, since they were implemented differently in different gateways and in different countries.

- Some added functionality was often requested by the customer to accept the product.

- The hardware and OS (HP-UX, Sun Solaris, Microsoft NT) differed, as did the database server, web server and application server software, thus the application had to be ported to a given environment determined by the customer.

- At the end of the installation, the testing of all the added code, new environments, localization etc. was required in order to minimize the risks of having a poorly functioning service.

Given all these modifications needed with a single customer, it was clear that they could not be achieved through parameterisation with reasonable development resources. A localization, customisation and tailoring / testing phase was also clearly needed. In the following figure a very simplistic description of the server application and its connections is offered.
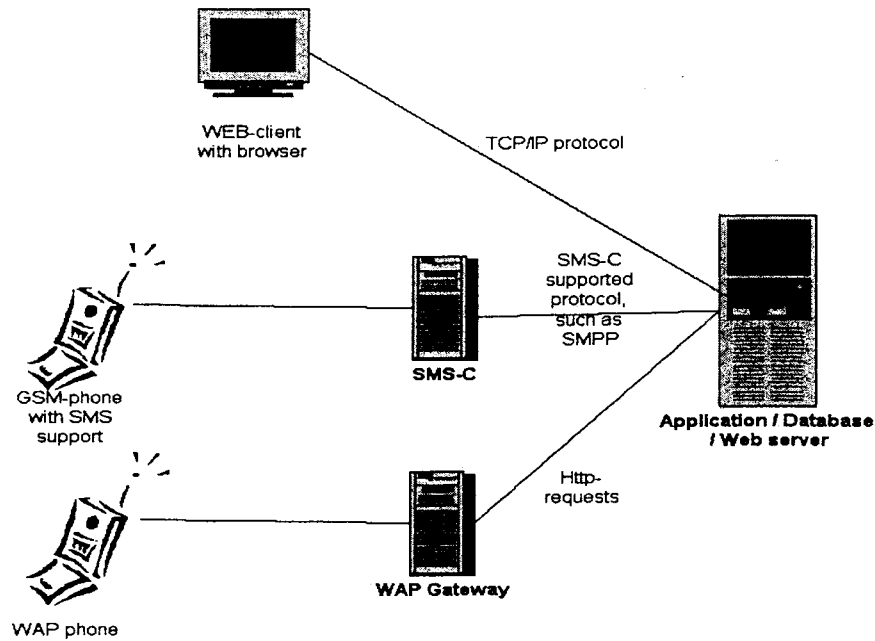
FIGURE 10 Basic architecture of the system

An additional thing that was also discovered that most teleoperators were rather complex organizations and the installation and testing of everything required the approval of several different parties within the teleoperator's organization, thus it was a time-consuming task to convince all the relevant parties that an access / modification of a certain part of the system, controlled by a certain division / unit was really necessary.

7.3   Internal processes and product development

Because X was still a relatively young company, X had it's processes yet undeveloped. One could characterize the development methods used at this time (Q1/2000) as ad hoc methods. Many of the tasks were organized in projects, which did not use any explicitly expressed development methods or other openly communicated structured approaches in development. Much of the work was not documented. Characteristic to the ad hoc tendency was that whenever there was a pressing problem, one of the 3 – 5 key coders was called in to solve it. This work included many "heroic" situations without the

knowledge being extensively leveraged within the organization. Almost all of the product ideas came from the development level. The role of total 1 – 3 salespersons was merely to find customers, make quotations and act as an intermediary between the development and the sales channel / customer. Sales personnel did not know how the products functioned as a whole; instead they could name a few key features of the products. At first this was identified as an acceptable status quo in the upper management, although questions were raised whether the sales personnel should be more technically competent. At this point also the understanding of how the end-user and the administrator interfaces of the software functioned were determined as technical issues.

The company had less than 10 products altogether, of which some already had working versions and some were on an idea level – all the products were intended for the teleoperator segment (or any similar segment being able to utilize such services with their end users). The majority of the products offering entertainment, but some were related to connectivity infrastructure and location based services.

Overall, one could say that there was very little marketing strategy involved at this point. One interesting requirement regarding the marketing strategy was, however, clearly ordered by the upper management. The software functionality was always to be made as generic as possible. According to the sales driven approach this allowed the same software to be used in very different contexts. For example an application controlling mobile billing transactions could be sold to control transactions in a timber mill. Upper management had a clear vision that the software could be sold to very different contexts with little or no modifications if it was generic enough. This was a very technical view, partially based on experiences from the Finnish domestic market where there was less competition in a single segment, which gives the vendor better possibilities not to focus on any specific segment.

After trying to run parallel projects during Q1-Q2/2000 the management of the company identified a clear problem. The development had serious problems. One of the biggest problems was related to resources. X still received tailoring tasks (which it could bill for) from earlier customers. The employees were, however, already allocated to different tailoring and product development projects.

When a new tailoring request came from a customer, the unit manager had a difficult decision to make, resources had to be taken out of either from other billable tailoring projects or from the product development projects to accomplish the task in a given time. In a very sales driven situation, it did not make sense to replace one billable tailoring project with another (annoying the first tailoring project customer already waiting for the promised results). On the other hand there was no fixed customer for the next release of the MOTS product yet, so one could not clearly identify from which exact customer and what exact amount of revenue would be lost if the product release was delayed enough to finish the required new tailoring task. To state the situation in a simplified manner, on the other side of the scale there was clear, certain revenue, achievable through tailoring and on the other side of the scale there was not. After a series of decisions, where the product development resources were always moved temporarily to a tailoring project it became evident that X was not able to carry out their product development as they had planned in the beginning of the product venture. The projects involved with product development simply were delayed too much by average. Later on the author of this thesis received feedback on similar results from several Finnish tailoring companies starting to focus on their product business, each having problems in allocating resources between tailoring and product business, the product business being the loser more often than not (SoftaPro 2001).

How to solve this situation that the company had run into? The approach to solve the situation was to gradually shift all the tailoring projects to the second unit of X and let the mobile unit concentrate on the product development. This public decision was carried out beginning Q3/2000 and it lasted until Q1/2001 to complete the execution. This is indicated in (Appendix 3). All this caused a clear change in the risk management of the company. Although there still was no clear company level strategy regarding revenue generating key products, marketing focus or geographical focus, there was now a clear decision that a certain expense level of running the whole mobile unit was accepted as a cost factor.

In general it can be said that entering an international MOTS field with the thought that this could somehow be accomplished with little risk, can be characterized as optimistic. At some point one has to have a well-productized software release available (with good

documentation). This requires resources, seldom fully funded by customers who have not seen the whole product yet.

7.4 The first, the mobile content product

As mentioned earlier, X had done some tailoring projects earlier and was able to show those results as references to other teleoperators and to the sales channels. During 1999 – 2000 there were not many well productized mobile entertainment products yet in the market. One of the sales channels was a VAR partnership with a hardware vendor H that also offered consulting. H had established relationships with several teleoperators, which enabled H to carry out negotiations on X's product (or a group of them related to mobile content). The product that the cooperation was mostly focused on was a server application enabling the users to download mobile content to their cellular phones. Due to the new nature of the field even some of the customers could not identify a large set of requirements. Instead they were more willing to adopt a large part of the offered functionality with the new, innovative product concept. Tailoring was needed to integrate the product to customer's existing systems. The following represents the business model of this examined case.
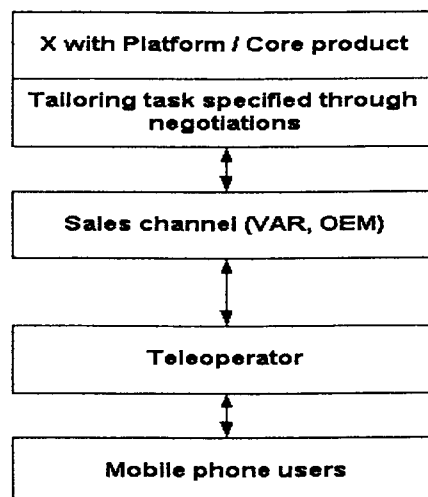

FIGURE 10 X's business model

X was able to close more than one deals with relatively poor, or nonexistent documentation of the product to innovator and visionary customers, who were willing to purchase the product based on demonstrations, without extensive documentation of performance measures, proof of reliability or internal architecture specifications. As the service was set up, running and generating revenue to the teleoperator, minor bugs and undocumented features were not the key issues when they were promised to be further developed in future releases. In more severe cases there were crashes of the server running the service. Since the product was still in its teens, innovated by young developers, it did not have any mechanisms of alerting the administrator when the system crashed (typically a Unix-server). This was seen as a nationwide denial of that particular entertainment service for the time period of the administrative staff to receive notice from others that there may be something wrong with the system. The warning system (or a backup system) was one of those features, which were not developed in the current release, but instead promised to be developed in future releases, as X had no experience of building such mechanisms.

A large part of the key sales responsibilities (with H) was carried out by people working in the product development. This was naturally due to a fact that since many of the issues were not documented, the technically oriented people were the only ones who could give answers and viewpoints on the issues at hand. It can be said that the customers at least partially understood the ad hoc nature of the situation. Although it must be stated that there was very little understanding regarding the rather expensive Unix server, third party application and web server software, the database server software and the software product itself all turning unusable due to malfunction.

As the mobile entertainment services started to become more common during Q3/2000 – Q1/2001, the situation in the marketplace changed rapidly. Even if the services were profitable to the teleoperators (thus there was a market demand), the MOTS products themselves were offered by a large variety of vendors. Given the relatively small physical size of the MOTS applications in question (a single application was in the range of 1000 – 5000 lines of Java code), multiple vendors were able to start offering similar services in relatively short period of time, thus causing the general price level of the products to drop significantly. Even so, no clear market leaders had emerged, so

there was no clear need to follow any specific company and X was still one of the few who could state having several actual installations running.

The biggest success so far had been accomplished with a group of products that enabled users to download entertainment content to their phones (this group was later combined to be a single product). This is discussed in (Appendix 4).

Regarding this product area a new requirement emerged from the market. The new requirement stated that operators wanted the application vendor to enable and offer professionally made content to be delivered to the operator, which in turn then made it publicly available to the end users. The current mechanism was limited to only allow the users of the service to create the content for themselves with the tools provided within the application. To fulfil this new requirement X would have had to start producing this professionally made content by it's own or negotiate a third party to do this. The software product would have needed tools to import and utilize such new data.

In this new situation the whole product definition had changed substantially. The original sales strategy of X was only to sell a product license (possibly through a partner), do a very short installation and not to have an active, intimate relationship with the customer afterwards. This was criticized within the management as being too close to COTS sales tactics, but was still the way the customer was decided to be processed. This resulted to the fact that X received very little feedback how the products (still in their early stages) should be developed. Now the new requirement took the demand of interaction even further from the present status quo. X would have needed to establish a monthly interaction with customers demanding to receive this type of content, which they were ready to pay for separately. Furthermore X would then have been drawn into a business of servicing customers with new content instead of selling software licences.

This type of setting seems very appealing when it is observed through the eyes of a MOTS company looking for ways to find out more about their key customers, becoming a key infrastructure vendor for a selected customer group and trying to establish an interactive communication mechanism between the buyer and the seller.

When observed through the eyes of a sales oriented company strongly focusing on licence sales the situation did not look appealing. There was no additional licence sales in sight (not counting the new customers asking questions of the same functionality) and the whole interaction with the customers would have needed a large amount of managerial work to create relationships with third parties and to create a business model satisfactory to all parties. An implicit requirement was to change into a different culture of constant interaction and servicing with customers and third party suppliers.

X made a strategic decision that it is not within their strategy to provide this type of content on a regular basis. Instead X will concentrate on the product and licence business instead. Customers seeking this type of full service were instructed to find the content (and related service) from other companies as X promised to support the imported data if there was a clear case for it.

X also suffered from technical problems with its products related to downloadable mobile content. Some of the installations had been running for a while and the teleoperators had suffered several crashes of the system. Regardless of the MOTS nature of the products X had not established working interaction mechanisms with the customers. In a case of a crash this caused severe problems since the requests for repair were often delayed and there was not any clear mechanisms established to address the remote situation. X lost two key customers during the process of trying to fix existing problems while simultaneously trying to concentrate on other focus areas (Appendix 5). The second lost case caused X some bad reputation. As the customer was requesting fixes to the system, which was suffering of weekly crashes the sales oriented response from the CEO of X was not to offer any fixes, but a clear indication that the customer would have to purchase a new release to remedy the situation. This can be seen as an indication of not having and not valuing the intimate relationships with the customers. Several customers presented threats of changing to another vendor if the quality of the product and interaction would not be improved.

As the time passed X was not able to find enough customers to keep this group of products profitable. The monitored product development costs were not covered by sales revenue. In Q3/2001 the company made a decision not to continue the technical development in this particular product field. A major competitor had emerged offering

also the required professional content through a subsidiary. This competitor had a revised business model of generating a large portion of its revenue from services and content, instead of just licence sales. This competitor was implementing an internal transformation process from a licence oriented to a service oriented company. Many of the customers saw this as the whole product they were interested of. There were also local tailoring companies in each country, which offered to implement similar services at a competitive price. X simply did not have a competitive offer to present any longer.

What is noteworthy, even as the X was an early beginner in this particular product segment, due to its decision of not to focus in any way, it never reached a position in any geographical or other market segment (such as the internet service providers) where it would have had an advantage of a know-how of the field or market leadership. Instead X was forced to find sales in all the product's segments always having to prove themselves as a new, trustworthy vendor without capabilities of pricing their products high due to an untrustworthy position.

Another issue is the role of product development. In this case this unit received very different requests from very different segments during the development period of this group of products. The product needed to function well as a standalone product, the product needed to be interconnected with different platforms, it also had to be able to import content from other parties etc. In more generic terms, to allow requirements come in from very different areas has a clear result. There is very little chance that the product development will be able to fulfil all these requests, thus possibly resulting in a chaotic situation in the PD / NPD. An example of such a situation with an ISP customer is described in (Appendix 6). This in turn can have a negative impact on the employees' motivation to develop company's development processes further. Comparing to a situation where the product development efforts are more organized and focused, this can act as a contingency factor to enable the development of the processes with good motivation of the personnel.

The negative circle closes when the NPD is not able to deliver different requests in time. This leads to a situation where the sales unit is even in a worse situation with customers and not only suffering from the lack of good market position. This can be seen as yet

another reason why segmentation and focusing is so important in the international software product business.

7.5 Internal development advances

As described earlier, X functioned according to ad hoc methods in the beginning. As more people were hired and more requests from different fields came in, some of the developers themselves started to ask for a more structured mechanisms to work with. The work related to quality and processes started in the second half of 2000 without any full time resources. During Q1/2001 a project was started with better resources to further formalize the development mechanisms. The waterfall model was chosen to be the first fully implemented development model. The start of the quality project is indicated in (Appendix 7). X started to move toward teams and roles of expertise, which did offer clearer responsibilities and enhanced visibility from the project management perspective. In Q1/2001 X already had formed teams developing their software product's newest version. These teams were able to utilize much of the implicit knowledge they had gathered during the development of previous releases. As the development grew more mature the testing of the results was clearly still the bottleneck. The results of a poor testing process are indicated in (Appendix 8). Testing the performance of requests passed by the system in a given time in multi-interface Unix server system is challenging even in the case of working internal processes and available resources, not to mention a case where neither of these factors are properly in place. Several different testing issues had to be considered: Code and its testing, testing of usability, documentation and its testing, testing of localization, testing of set-up and testing of performance. It was experienced more than once when a project date was set to deliver the next version of a product that the code was "ready" at that time, not any other issues. In these cases it was often found that after a round of testing the readiness of the code could also be questioned. The maturity of internal processes had clearly improved regarding the development. However, to be able to execute the whole chain of actions from an idea to a finished, tested product, a very extensive knowledge of the whole process must be created within the company. In this view, the maturity and

ability to execute the overall process was not yet sufficient. These issues were a matter of future development.

7.6 The second, mobile communication product

Simultaneously to the developments in the case one, another product was also under development in X. This product was a server application, which enabled mobile communication between the users of different mobile and Internet terminal types. As the first case had its origins in the earlier tailoring business, this second product started with a strong partner P, who originally requested such a product to be developed for sale by their international mobile software product business. On the contrary X had negotiated to develop the product as their own and was thereby responsible for the business risks developing such a software product. X also saw the product in question to have a lucrative future.

7.6.1 The early phase – release 1.0

The development had already started in Q1/2000 and given the ad hoc nature of X's development, the project was delayed and it took till Q3/2000 when the release 1.0 of this mobile communication product for the young generation (thus targeted for the same end-user segment as the first product) was finished. X was again very early in the market; there was very little competition from other software companies with well-productized offerings in the beginning of the year. The signed partnership agreement offered strong marketing assistance, which so far had been the weak side of X. The downside was that since the partnership was of OEM type, X did not get any visibility even in the case of closed sales, in fact the partnership agreement prevented X from using these sales as reference. Instead, P sold the product under its own name.

Release 1.0 proved to be a disappointment – the product concept was not well thought out, X did not have any resources dedicated for product management or product concept creation. This led to a situation where the developers were (again) greatly responsible for the concept development of release 1.0. It took more than 15 iterations of releasing

the software and P testing it and coming back with multiple bugs before the quality of the release 1 was of satisfactory level. The communication of this type of iteration is described in (Appendix 9). The original idea behind P's testing was only to assure that X had done good quality and relevant testing. It turned out in the process that suddenly P was forced to test the results, since X's quality was not of satisfactory level. As P had a geographically distributed organization, this extra task caused P's testing budget to multiply from the originally planned. Some bugs were conceptual, meaning that the software behaved as the designer of the software had intended, but the functionality did not make sense from the customer perspective. Some bugs were more of the traditional type, where the functionality was not as planned. X's own testing was only beginning to emerge.

This release was introduced to several teleoperators (even as a pre-release) of which some decided to trial the product. This is again noted in (Appendix 10). All turned the product offering down during Q3, without any actual closed sales. Some reasons for this were lack of good quality documentation, poor product concept design, poor usability and a relatively small set of functionality. Following the results P's management expressed concern of the progress of the development and marketing efforts.

At the time of finishing release 1.0 product management resources were added and a quick decision was made that X will speedily develop an enhanced release 2.0 which will be finalized in Q4/2000.

7.6.2   Release 2.0 – facing a more mature market

Much more conceptual work regarding the release 2.0 was done compared to the release 1.0 project. This involved redesigning parts of the software's user communication mechanisms, analysing what users found comfortable, understandable and what they did not. This work was led by program and product management. The development team and its project manager participated heavily on this work. It is notable that the implicit knowledge of the domain increased significantly as this type of conceptual work progressed. This greatly improved the possibilities for the personnel involved with the development task to further suggest feasible improvements. It is suggested that this type

of knowledge generation can be a very generic phenomenon when developing innovative software products. Similar kind of development was seen also in other projects outside of this study. If this suggestion is accepted, that would indicate a very large process contingency factor established from the use of product concept teams having the possibility to concentrate on a specific area for a longer period of time, thus lifting the amount and the quality of implicit knowledge, which then can be transformed into codified products. Comparing to Netscape and Microsoft (Cusumano et al. 1998, 245-254), both organizations seem to deploy product-specific teams and product management to innovate and specify the products as their selected way of organizing the product development.

One new process proved to be a difficult one to learn. Since the cooperation was of OEM type, the management of P wanted to understand all the new features and functionalities prior to the new release of the product. P used this information for communicating the new qualities of the software internally within its own organization. This activity targeted the pre-sales activities of the new release. For X this meant the responsibility to communicate everything (and commit to it) prior to having much of the functionality implemented or documented in a formal way. This process was important since any functionalities missed in this communication were also missed by P's organization and by P's customers in the first round of contacting. A constraint felt in the product development was that if there were any good inventions created while developing the product, any prior communication regarding the product's future functionality could prevent that invention from being implemented into that specific release. As X simultaneously preserved the right to use the direct selling mechanisms for the product, they had a motivation to design the product as appealing to their (similar) customer segment as did P. Sometimes this meant conflicting interests, which were a matter of negotiations.

When finalized in Q4/2000, the release 2.0 was introduced to several teleoperator customers by P. This time the product received good reviews from customers and first deals were closed. This meant that X got to practice its delivery mechanisms and the quality of those processes. Unfortunately the localization and customisation of each installation took far more time than originally planned. This type of situation is

illustrated in (Appendix 8, 11). The installation schedules moved from the level of two weeks to even six months of finalizing the installation in worst cases – in which time the commercial side got complicated and started causing delays also. Another indication of improvement was received in the area of performance measurements. During the early market these figures had not been very important, as long as X had promised the performance to be "sufficient", the customers had not seen this as a major problem. In the more developed market the performance measurements were a must for some customers. Unfortunately X had no experience of demanding multi-interface server application's performance testing, which caused delays with some customers requesting for detailed performance measurements.

As the deals were being closed during Q1-Q2/2001 the competition started to increase. Several companies from middle and southern Europe were receiving significant Venture Capitalist-funding (VC) and were developing similar solutions with well-tuned product concepts and substantial marketing efforts. All of these companies seemed to have a much more focused product approach. They defined the user group for their communication applications to be exactly 15 – 25 year old end users, the customer focus being the teleoperators (not internet service providers or application service providers) and the whole functionality of the software, documentation and product concept was targeted to address this particular segmentation. They also were targeting their sales in Europe at first, instead of spending their resources in Asia, America(s) and Europe simultaneously. An additional difference in some competitors approach was clearly the time they were spending with the customers and different parts of their organization. At least some of the competitors (there is no exact evidence on all of the companies in the field) spent large amounts of time with the teleoperator organization and established multiple personal relationships with different people working in different sectors of the customer organization. This enabled them to participate in the planning of the mobile services of the specific customer, as well as to receive immediate feedback of what the customer saw important. An important factor seemed to be the implicit understanding of what the key customer personnel actually valued and was interested in. The competition was able to demonstrate this in seminars, where they had other developers and the customer teleoperator to attend to exchange views of what

would be commercially interesting in the future and what kind of cooperation they (competitor and a teleoperator) had had in the past.

As these competing companies quickly reached sales in their domestic and nearby regions, the market was clearly changing for this product segment during Q1 – Q2/2001. The price level of a single application dropped to less than half of what had been the realized price of the first sold installations of X. Unfortunately the competition had already created and implemented a revised business model for the new situation. All of the competitors having multiple references suddenly showed a similar approach: They had created other modules / applications to be offered *with* the original communication product. Sometimes some of the more advanced functionality of the original products was moved to these additional modules. These modules related to different themes of communicating and offered extra functionality to the original product. This enabled the competition to price the original product cheaply and still maintain a relatively large deal size as they were selling also other components with the original communication product (naturally with separate prices).

X was developing a release 3.0 for the product to gain some of the product's concept level advantage the competitors had gained with their new releases during Q1/2001. The release 3 was given some very challenging requirements. As the market was clearly moving to a more mature state, extra small features were not seen to help the competitiveness of the original product. Instead it was decided that release should have full scalability features with redundancy. This meant that the application should be able to run on multiple servers (for example a Unix-cluster) simultaneously and to be able to handle a single server breakdown. This type of functionality was believed to address the requirements of the teleoperator segment by P. X agreed even knowing that this requirement was beyond their current level of expertise. However, some of the young, relatively experienced developers indicated that this type of implementation was "within technical limits." However, they were not able to give any certainty of the timetables. The plan was that when successful in this, the ability to price high would be reinforced. To accomplish the plan the BEA Weblogic application server environment was decided to be the new running environment for the application since BEA offered ready functionality to accomplish scalability. The product itself was decided to be designed

according to Java's EJB-specification in order utilize BEA's clustering capabilities. This kind of robustness was seen of interest to the teleoperator segment in question. After developing the new release during Q1 and Q2 /2001 it was clear that a well-productized release of 3.0 was still far away. P had originally been promised a release some months earlier, but was delivered only pre-beta level versions to experiment with. As it seemed that the release 3.0 would be postponed too far, it was jointly decided that a release 3.0 would be published with some new features, without the scalability and redundancy during the summer 2001. This is again noted in (Appendix 12).

By the end of Q2/2001 X had lost its momentary advantage in product concept and feature level comparison. P communicated in a meeting that more sellable modules, resources and focus to the application area must be allocated with a new release 4.0, since they did not find bug-fixed releases 2 or 3 sellable on a large scale any longer. This was caused by the weak position in the negotiation table when confronting competitors with multiple modules and a low price on the original product.

Internally X carried out negotiations how to include ASP-segment functionality to this product, since it had already sold one licence (through another sales channel) claiming that this functionality was already "under development." It was estimated by the product management, that to reach a whole product status in this ASP customer segment the development would be needed to carry out at least a 12 month project to reach this goal. It was also suggested by the product management that more requirements might arise from this segment while the work progressed. This occurrence was aligned with the sales oriented strategy of not letting go of any sales opportunity, or segment.

In Q3/2001 several developments occurred in parallel. P continued to request X to develop a new release 4 with scalability, redundant connectivity, enhanced product concept with multiple modules and enhanced usability. P emphasized, that the current release was no longer competitive and that the teleoperator segment they were solely targeting needed much faster development cycles and more resources in the product development. The different requirements are noted in (Appendix 13). These negotiations were time consuming (lasted several months) and were difficult for X, since they did not have much leeway to offer many of the things P requested during these negotiations in a short period of implementation time. Additional sales in large

extent with the current software release seemed hard to accomplish. X was spending resources to fulfil the promised ASP / ISP requests (had been since Q1) in a clearly understaffed position to reach good quality results in either of the segments. The resources put into the (still internally unsegmented) communication product were constantly indicating to the X's program management that the technical quality of the solution was of very poor level since there was not enough time to design and implement all the required features in an orderly fashion.

In Q3/2001 the product management was partially allocated to assist with sales instead of being able to concentrate on concept and feature development. This was again a very sales oriented view; the future sales of the product were at risk due to the recent poor market success. After negotiations with P it became evident that X was not going to be able to satisfy the requirements P had during 2001, as was the request. The existing ASP customer requested their money back from poorly working software and was turned down with their request. During Q3 P announced the discontinuance of the product and allocated all prior resources to other tasks. Officially this was communicated to be part of a larger restructuring of non-profitable businesses.

During 2001 X was not able to close any licence deals with this product through direct sales and reorganized its product development to a tailoring unit. From the beginning of 2002 the tailoring unit was communicated to do product development for the communication product when there is a clear buyer for a new product release. Otherwise the old developers were to be allocated to billable tailoring projects. A more marketing oriented restructuring took place from the beginning of 2002 as X added its own sales personnel to increase the efforts in direct sales of software products. In the annual statement of 2001 of X it was declared "the company will invest especially into sales and marketing." According to the same annual statement of X, the turnover of the X during 2001 was little over 2 million €, the loss of the company was equal.

8 THEORETICAL CONTRIBUTIONS - DESIGNING THE FRAMEWORK FOR THE MOTS PRODUCT COMPANY DOMAIN

In the theoretical chapters (2 – 6) of this research a theory foundation concerning the product and tailoring domains was introduced. In this chapter the data gathered from two sources (case and theory) are analysed and combined to reach a model, which can serve as a framework for the development from tailoring to successful MOTS product orientation.

In the case examined in the earlier chapter, two separate products were studied. Both of these products were intended for the international MOTS software product market, the teleoperators being the main target group, but not the only one. The first examined case was utilizing a VAR model and the second product was utilizing the OEM model. In both of these cases during two years of examination, company X had a very favourable position of being well financed and receiving plenty of marketing support and cooperation from its partners. Another issue in common for both cases was that X was early in the marketplace offering solutions which very few significant competitors (if any) could offer in the beginning of the product / project. Furthermore, both of these very new product types became widespread, common products utilized by the teleoperators worldwide.

Robert G. Cooper has examined the reasons for product failure. Many of the reasons he identifies can also be identified in company X's case study. Cooper identifies the following reasons (Cooper 2001, 23-24):

- *Poor marketing research.* Insufficient or faulty marketing research is what managers cite most frequently as the number one cause of new product failure: "A lack of thoroughness in identifying real needs in the marketplace, or in spotting early signs of competitors girding up to take the offensive, is often the finding of a new postmortem."

- *Technical Problems.* The second most common cause of new product failure is technical problems in design and in production. [...].

- *Insufficient marketing effort.* Here management is guilty of "assuming that the product would sell itself" and simply failed to back the product's launch with sufficient marketing, selling and promotional resources. [...].

- *Bad timing.* Timing issues surface as a key reason for failure, not only in these studies but in countless others. The penalties of moving too slowly, or too fast, stem not only from technical problems but also from flawed planning, organization, or control. Numerous new product failures result from not moving quickly enough, given a limited window of opportunity. [...].

Another list by Cooper identifies the following reasons for failure as follows (Cooper 1999, 119-129):

- *Ignorance: We Don't Know What Should be Done.* Some companies' leadership teams simply don't understand what's required to make new products successful. That is, they lack a complete and balanced perspective on what a well-run project looks like – what the important tasks and events are. [...].

- *Lack of Skills: We Don't Know How to Do It and / or We Underestimate What's Involved.* [...] Some of the skills include, for example: How to undertake needed market studies (user needs analysis, competitive analysis, concept testing, segmentation and market analysis, etc.). How to do the up-front homework, build a business case, and undertake a business and financial analysis. How to design an effective launch plan – both operations and marketing. How to run projects [...]. How to deal with joint venture and alliance partners [...]. A knowledge of the technology required to design, develop and produce the product. How to lead an effective cross-functional team – conflict resolution, communication, etc.

- *A Faulty or Misapplied New Product Process.* More than half the companies today claim to have a new product roadmap or process in place – one that guides them from idea to marketplace. The trouble is, the process is missing key elements and / or it's poorly applied.

- *Too Confident: We Already Know the Answers.* "Why bother doing this product test or this market study... we already know what the result will be." [...] A lack of good market information and inadequate homework are cited consistently as the number one reasons for new product failure! [...].

- *A Lack of Discipline: No Leadership.* [...] One of the problems in product innovation is that many of the prescribed actions are *discretionary* or *optional.* [...] Often the lack of discipline comes from the top. The leadership team of the business "talks the talk", but doesn't walk the talk. Indeed, the leadership team is often the first to break the rank – to break discipline. [...].

- *In Just Too Big a Hurry.* "We've got some tough deadlines... a limited window of opportunity. This product must be launched for the trade show in September. So we're going to have to cut some corners – cut out the market studies, cut short the product tests, do a quickie sales force training effort – in order to make that deadline. Let's just get it out there!" [...].

- *Too Many Projects and Not Enough Resources:* A Lack of Money and People to Do the Job. [...] This is the result of two management failures: management doesn't provide the necessary resources to achieve the business's new product goals; or they approve too many projects for the limited resources available. [...].

Comparing to the studied company X it is difficult to confirm, that any of the identified factors would be missing completely. As the Cooper's identifications are listed here, the analysis of the case study and theory is continued.

8.1 Analysis of the business strategy

X clearly did not have a clearly planned, explicitly documented and communicated product or company strategy. The statements made in this area by the upper management did not answer questions what, how or to whom the company is going to sell in the sufficient accuracy. A list of all the possible segments for the product(s) does not qualify for a detailed product strategy in the light of the theory presented earlier in this thesis. It is notable in the studied two-year period that the sales driven decisions

regarding the product's generic nature actually prevented the focus required for the product's focus success factor. This type of behaviour may be acceptable in the very early stages of the product, if no feasible direction is found. The situation is however pressing and turns soon into very problematic, when the software company is losing the markets when they are formed and more domain-oriented, focused competitors are taking over.

This problem also had its effect on the whole organization and its units. The product development had clear difficulties satisfying different requirements from different segments. Also the sales and marketing wasted resources on different segments and were not able to focus their actions on any specific product or segment. This in turn hindered severely the development of explicit and implicit knowledge of any specific domain in the whole company. The issue of selecting a segment, which in fact is a worldwide industry can also be criticized. For a small or medium sized company, it is recommendable to start with a much smaller segment to work with. Geoffrey Moore (Moore 1998, 100) suggests that if the target of the company is to reach a favourable market-leader position (controls more than 50% of the software product segment) it should not target a market segment bigger than the expected amount of the annual turnover of the software company itself. For example if the turnover of the company is 10 units, it should target a software product market of maximum 10 units of product-specific revenues annually. For too large segments the subsegmentation is recommended (Ibid).

As the management of X did not succeed in selecting any successful focus for the company, another level of thought was also left somewhat unattended; selecting from operational excellence, customer intimacy, and product (or service) innovativeness was not done. Clearly X made decisions against customer intimacy and it did not encourage long on-site periods of staff on customer premises. Given the maturity of X's processes and the staffing of the product unit, choosing operational excellence could have been problematic. As the X's staff was young (as was the target market), selecting product innovativeness as a strategy can be seen feasible, but was never selected or resourced appropriately. Not being able to select and implement any of these options can also be seen as an extra hindrance to X's market development.

Many of the reasons causing X not to succeed with its products seem to be caused in the very early stages of the company's product orientation. The decision of not to hire any internationally experienced upper management can be seen as one of the major contributors to X's development. One would require the top management to have experience from designing effective business models in the international environment, have plenty of experience in product-oriented managerial tasks, have experience in international sales and marketing and to be able to craft a detailed, feasible product strategy before this type of product venture can be seen as sufficiently staffed by the upper management. In the light of this case study, the experiences from the tailoring field do not necessarily support these requirements.

After the management have been staffed and the product focus (or the non-existence of it) has been decided and the company's strategy is decided and implemented by the selected management, the company is locked into functioning within the limits of its resources. If the implemented strategy is then a very sales oriented one, causing the workload to multiply when trying to reach the whole product status in all of the selected segments, the limits of the (domestic-experienced) resources are reached and exceeded very quickly.

In the light of the case study and presented theory it would seem rational for small and medium sized tailoring companies (and any ventures with similar resources) to consider less ambitious MOTS product options than full globalization at first. This would allow them to start the learning process of product orientation and product markets for example in the domestic and some nearby (such as Nordic) markets, thus exercising geographical subsegmentation (Autere et al. 1999, 38), (Moore 1998, 37-38). This type of a start can greatly reduce the time and money spent on marketing, sales and installations in different continents in the later phases of the company's journey. This would seem contingent, since the small MOTS company / venture cannot assume any sales channel to handle the sales, customization and installation of the product(s) in an autonomic way in the beginning, thus requiring the software company to invest significantly in every new installation in a strange environment.

If it really seems that there are strong arguments in favor of approaching directly the wide international market without having sufficient time to practice the area, a very

clear strategy, strict product focus and clear segment selection is strongly recommendable to give a definite direction for the company. The non-segmenting, generic product strategy should be exercised only domestically, if at all.

8.2 Analysis of the internal processes and development work

The situation of a tailoring software company starting MOTS product business is naturally dependent upon its former processes and their quality. If the internal development processes have been mature and of good quality earlier, adopting to the product business can be internally accomplished more easily, especially regarding the development work. In the case study X did not have its development processes in a mature level prior to entering the product business. At first this did not seem to have a great significance, since the visionary customers were willing to overlook new, innovative product's faults and did not set additional requirements in very large extent. As the market grows more mature, the requirement for the internal processes to provide deliverables (such as new versions, tailored solutions, documentation) and quality is increased significantly. If this is well understood by the management, there is time to practise these issues. When the increased requirements are already presented by pragmatist customers, it may be too late to start the process development work to improve the productivity and quality of deliverables of the company. When understanding the market development, the management of the company should start this type of development in the very early stages.

Careful planning must take place to define the additional required tasks, processes and resources in a product company, case by case. Internally technical development areas requiring more work (compared to tailoring) concern requirements analysis, documentation, version control and testing. Among other areas requiring more work is the new role of sales and marketing, the highly enhanced role of upper management, product management and their strategy work in different areas. (Compare to FIGURE 4 The market-driven planning framework (Räsänen 2001) on page 35, which outlines 7 different strategy areas to be designed). This level of holistic planning did not exist

within X, which led to situations where the different units and their goals were not aligned (such as product strategy, quality strategy, sales strategy).

It is notable that only the separation of product development, product management and marketing and sales from the "single core group" of personnel is a very large management task. To accomplish this, new resources, reallocation of existing ones and new interaction processes are required. The successful and satisfying interaction between the product development unit and product management and marketing unit can be seen as a clear quality attribute for the processes of the software company, especially in the case of a more mature software product company. As long as the company is able to establish and communicate a commonly accepted product strategy and based on this is able to carry out the release planning of the products in good cooperation between the parties, it can be assumed that the products take into account different perspectives in good manner.

The development process is a single domain needing an experienced, technically competent person to lead it. Company's coding practises, version control, used libraries, overall architecture of the products, internally built components and their use are issues requiring managerial work and some degree of quality system built around them to function effectively. A long practising period, plenty of training and constant managerial involvement are required to reach a mature state in this area. Having a person employed who has experience of this field in the product context can significantly contribute to the speed of reaching the maturity in this particular area.

In a MOTS product environment the ability to produce prototypes of new releases is valued by customers the pre-releases are demonstrated to. Another contingency is the ability to include new features and innovations to the product release in late phases of the release development. This type of setting suggests a flexible, possibly iterative, or more prototype-oriented development process to be used, instead of the waterfall model.

As the tailoring unit tries to transform into product-oriented unit the requirement to define (and test in practise) the new responsibility areas is imminent. There needs to be an entity responsible for the company strategy and an entity responsible for the product strategy. Internal processes should determine how these responsible entities (either a

group or a person) receive the product-field data relevant to their work and how they can effectively have a bearing on the product's features and overall focus. If these tasks are allocated to the middle management (such as product management), then there should not be too much interference from the upper management regarding the product decisions. On the other hand, if these tasks are allocated to the upper management, then it should be made certain that the upper management constantly receives the relevant data (e.g. the process supports this) to enable them to make product- and possibly segment-specific decisions. This is an emerging issue when the software company grows, new roles are established, and there cannot be any single manager / director, who is responsible for almost everything.

8.3   A framework for companies starting MOTS product business

As the theoretical part and the case study have shown, there is a multitude of issues and their interrelationships that can go wrong during a company's venture in the MOTS product business domain. What has been missing in the literature is a holistic framework, which would enable the young product company's management to identify significant areas of interest and to pinpoint their interrelationships with each other. It seems that a vast majority of consultants, specialists and such are mostly examining their area of expertise, but the company's management can often find it difficult to sort out the importance and interrelationships of different issues in different phases of the company's life. Very often this kind of work is done under great uncertainty of where and how the company should concentrate on.

In the light of the theory and the case the U.S. market and Europe seem to have a distinction in the young software product company domain. The workforce market in the U.S. evidently supports startups and ventures to launch their new product business and specifically to hire experienced, even specialized personnel, including managerial personnel for the new product ventures. One factor for this is probably the sheer size of the U.S. workforce market. In software product business in Europe the situation seems to be different, especially in smaller European countries. The workforce market of software product specialists of a single product area of a European country is typically

relatively small. This causes a situation where there may not always be specialized personnel available for a new software product venture, as was experienced in some situations by X (in other cases the decisions of not to hire were more intentional). This suggests, that growing domain-specific expertise first through tailoring can be a feasible approach in many European countries.

To be able to condense the MOTS product company's development path and its relationship with the maturity of the company's processes, a stage growth model has been created by the author of this research to clarify the development process of the company. This model is the major academic contribution of this thesis in a rather unexplored area of transformation from tailored software production to successful MOTS software product business. It is noteworthy that according to the case study and the theory of approaching the market on page 50, the required ability to produce the different deliverables differs considerably at different phases of the company's life. This development and the focus areas of different phases of the model can be seen as a key high level areas for a MOTS product company's management to work with.

It is notable that almost none of these developments can be realized without careful planning, a large amount of managerial work and a competent, well-motivated staff to participate in the process. Even when these issues are placed on a model one after another, it does not mean that they will take place automatically. Behind the model lies a clear assumption that qualified resources are allocated to carry out these tasks. The inability to reach the targets described in the model can very well be due to missing resource allocation, incompetence, inexperience, bad motivation or other reasons. In the model, advancing from one phase to another requires plenty of hard work and intellectual people allocated to design (and carry out) the exercises required. If some of the required components are missing, this can easily lead to a situation where the company's advancement is hindered severely. In such a case the company should either reallocate existing, or seek for new, more competent personnel to be placed responsible for the development of the issues described in the model. The presented model has both normative and descriptive qualities; the Business Strategy level and the Customers setting requirements level more of normative nature, while the Internal Process level and the Software Product Development level are of descriptive nature.
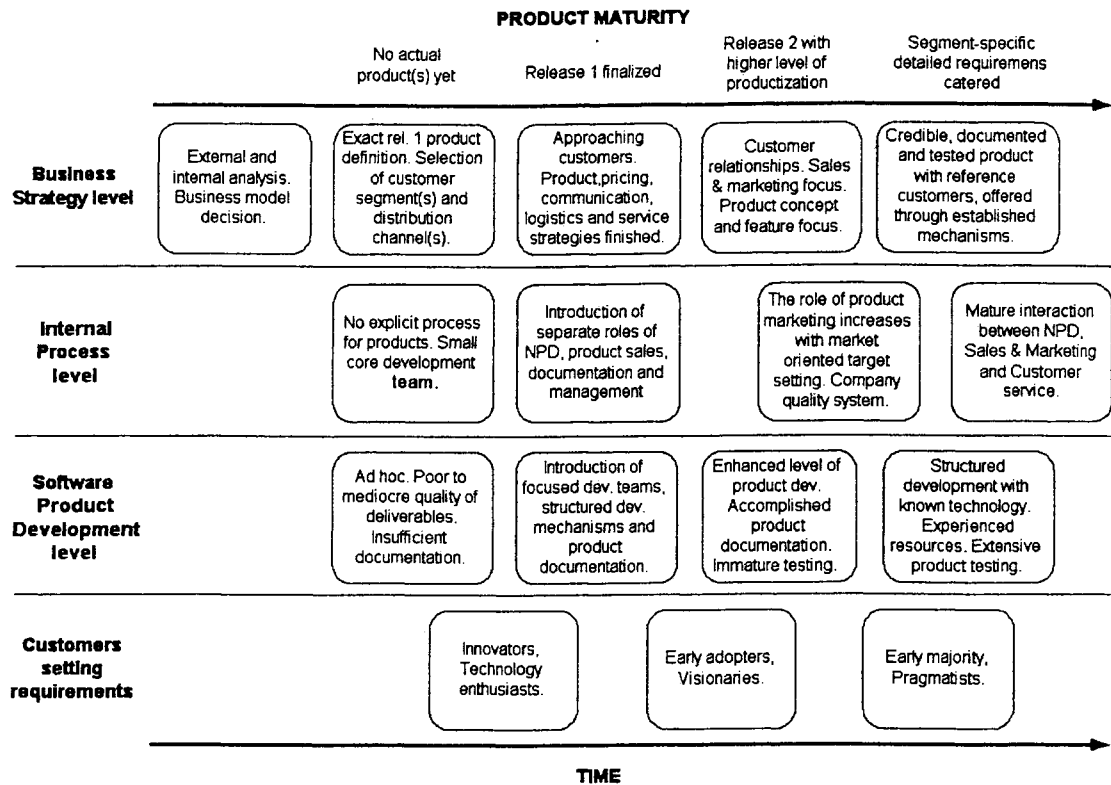
PRODUCT MATURITY

	No actual product(s) yet	Release 1 finalized	Release 2 with higher level of productization	Segment-specific detailed requiremens catered
Business Strategy level	External and internal analysis. Business model decision.	Exact rel. 1 product definition. Selection of customer segment(s) and distribution channel(s).	Approaching customers. Product,pricing, communication, logistics and service strategies finished.	Customer relationships. Sales & marketing focus. Product concept and feature focus.	Credible, documented and tested product with reference customers, offered through established mechanisms.
Internal Process level		No explicit process for products. Small core development team.	Introduction of separate roles of NPD, product sales, documentation and management	The role of product marketing increases with market oriented target setting. Company quality system.	Mature interaction between NPD, Sales & Marketing and Customer service.
Software Product Development level		Ad hoc. Poor to mediocre quality of deliverables. Insufficient documentation.	Introduction of focused dev. teams, structured dev. mechanisms and product documentation.	Enhanced level of product dev. Accomplished product documentation. Immature testing.	Structured development with known technology. Experienced resources. Extensive product testing.
Customers setting requirements		Innovators, Technology enthusiasts.	Early adopters, Visionaries.	Early majority, Pragmatists.	

TIME

FIGURE 11 Stage growth model from tailoring to MOTS software product business

Explanation of the model:

Business Strategy level

Starting from the upper left corner: First the company's management must assess the internal capabilities and resources and compare them with the market circumstances and possibilities. The closer the match between the compared areas, the better are the possibilities to proceed with less complication. At this moment there is no technical product development work required for the purposes of the process.

In the next phase several areas of decision-making are interconnected. At least a clear definition of the software product is required; this affects the selection of customer segment, which in turn affects the selection of distribution channels. In turn the

customer segment may have some influence on the product definition, as may also do the distribution channels.

The third phase of business strategy level is to finalize the work regarding strategies in the pricing, communication (how is the interaction effectively carried out with customers before, during and after the license sales), logistics and service areas. Close to this time period the company should also be able to deliver their first software product release to be introduced to the selected customer segment and receive valuable feedback in trials and other customer related situations. Similarly, close to this time the first sales with first real customers should be accomplished (if not done earlier).

After finalizing the third phase, the release 2 of the software should come out with higher level of productization than in the first release. For company's management this is the time to take good care of the existing, valuable reference customers, as well as to be able to focus on the selected segment's clientele who may be more willing to accept the release 2 of the software. This is the time where the requirement level of the customers is starting to increase, which means that the sales and marketing is required to have a functioning process with a clear case to present to new customers. The accuracy of product concept and features is essential, since the customers may not be willing to accept poorly documented ideas anymore. There is also bound to be increasing competition at latest by now (if it has been lesser earlier). With release 2 the software company should be able to reach additional sales if the product focus has been correct in relation with the selected segment.

In the final strategy phase the management of the software company should focus and prepare them to be ready to offer their product to the pragmatist customers of the segment. The requirement level of this customer group is very high and often many of these customers are also the biggest customers in the segment. Focus areas (which may not yet be at a required level) for the management are documentation, testing, reference customers and the good functioning of selected sales channels / direct sales.

Internal process level

This area is also under constant development during company's advancement to software product business. It is notable that regardless of the tailoring company having good internal processes earlier, the company must establish new ones to satisfy the different business process requirements of the product business. New experienced resources can be seen as beneficial to improve the situation. A typical starting situation is that a small core development team is formed in the very beginning to produce a working software product release. This may happen with little formality in the development methods.

The next phase is to introduce more staff to work with the product(s). At this stage it is no longer possible having everyone being experts in everything. The sales responsibility is allocated to full-time personnel. Similarly, the amount of documentation required is of the level of one or more full-time persons. Since tasks are being divided between specialized personnel, there needs to be more management to coordinate and improve the process and all the related tasks. Also clearer responsibility areas are starting to form (partially) due to specialized tasks.

In the third phase of the internal processes, the role of the product management increases (in more mature organizations this can take place earlier). Since the product concept and features have become the key competitive tools, it is no longer possible to succeed with only good project management and general sales personnel. As the company has been already developing its processes for some time, the first system describing the roles and interaction of different units of the company is maturing. This can be described as the first quality system of the company, either formal or informal. If the staffing and their motivation have been successful, this new internal process is very product oriented, instead of a tailoring-oriented one. The clear goal of this set of rules is to guarantee the rational, efficient functioning of different parts of the company to create quality products with strong market and user orientation. At this point the ad hoc world is left behind at large part.

In the last phase it is useful to analyse how well the different units of the company actually function together and create added value for the company efficiently. As the

quality system often can relatively easily describe the processes within a single unit (such as the development process), there can be situations, which are difficult to foresee and require the cooperation of several units (sudden key customer requirements, failure of one unit accomplishing the allocated task, etc.). It is after this level of cooperation is accomplished when the software company's internal processes can be considered competitive against other international software vendors.

Software product development level

These steps are described assuming that the former tailoring company did not have a mature process for its activities. This is a situation often encountered in young tailoring companies. If the company has had a longer experience and possesses already tried development mechanisms, it can pass the first phase of this level easily, although the new product documentation is yet to be developed regardless of the company type. In the first phase of a younger company the development methods are typically ad hoc. With little testing the quality of the deliverables is relatively poor and the documentation is yet to be developed.

In the second phase the more structured development mechanisms are introduced - this is a step that may already be accomplished by a more mature tailoring company. At this phase the requirement for good documentation is raised by sales (they cannot do their work if they don't have something to show), thus the maturity of the product-focused organization is lifted in the documentation area. This includes planning what kind of documentation is needed by all the related interest groups, creating policies regarding the usage and distribution of the documents, designing the knowledge transferral from development to documentation, designing document structures and ensuring the quality of the document process as a whole. As the company's staff exercises their new, structured development work, also the teams are beginning to form (either formally or informally) for each product area.

In the third phase of software product development the unit in question is able to function in a more certain, predictable way after the new development mechanisms are in place. At this stage also the product documentation has been practised a sufficient period for it to exist and have a relatively mature creation process. As the development

as a whole has matured into an appropriate level there is often one area of work left which few talented persons volunteer for, the testing of the products. As the complexity of the product(s) and the number of customized installations increases, the testing requires at least as talented personnel as does the development. Typically this area is resourced low and with inexperienced personnel in the early phases of organized testing, resulting to a situation where the maturity of this area is accomplished among the last areas of development work. Unfortunately this suggests that the overall technical quality of the product(s) will not be very high at this phase, even if the actual design and implementation of the product(s) has already reached a relatively mature level.

The last phase to be accomplished is to arrive to a situation where the used technology (programming languages, databases, operating systems, customer integration points, etc.) has become well known to the development staff of the company, hence minimizing the required level of extra learning when exercising common tasks. When this is combined with the implemented development mechanisms the development can be expected to deliver the required deliverables reasonably in time by average. Also the testing should reach its maturity at this stage. This ensures the quality of the deliverables to reach a satisfactory level, concerning also the documentation.

Customers setting requirements

In the very beginning the quality of the products is not sufficient for all the possible customer segments. The customer group most likely to accept a new software offer is the companies whose management is innovative and/or technologically enthusiastic. This group is more willing to look away from issues such as bugs, missing features and poor documentation.

The next group the company can advance to is the early adopters and visionaries. This group clearly sets more requirements on the performance of the software company, but does not yet demand a fully productized software solution.

The most important group to conquer (regarding the revenue from the product) is the early majority, the pragmatists. This group often controls the purchases of large,

established corporations. They require a high level of productization and a very good quality of interaction between the two companies. This requirement level can only be accomplished if the software company has matured its processes.

8.4   Analysing the presented model

Based on both experience and the theoretical part of this study there are two fundamental conclusions drawn regarding this model:

1. In each level the tailoring company must proceed through different phases in order to arrive to the position described on the right side of the figure. Each phase builds knowledge and know-how on top of the former phase. The last position described acts as a contingency factor for the software company to succeed in the selected segment / mainstream market. Some phases may be easier to pass if the tailoring company has had a mature process to begin with (especially in the Software Product Development level).

2. The probability of the company *not encountering* any severe difficulties in the examined area is enhanced if the company follows each level of the model *in parallel* to the other levels. This suggests that it is beneficial to develop the company and its targets holistically, instead of trying to focus too strongly in the progress of any single level, causing other levels to come short in their ability to execute their required tasks.

It is useful to analyse some common situations possibly encountered in software product oriented companies and reflect them with the presented model. First the issue of advancing unparallel with different levels of the presented model is discussed.

Advancing very far in the **business strategy level** has a clear implication. The management of the company is not accomplishing market (or sales) oriented tasks. Another risk with this approach is that the strategy should always be monitored and adjusted when real-life occurrences indicate so. There is little point doing far-reaching strategy work, if the basic assumptions underneath are changed by reality before the execution of the strategy takes place.

Internal processes. It is difficult to imagine a detailed, company-wide implemented process, which would be useful if it is not firmly based on strategy, customers or software product development work done in reality. These types of theoretical processes are very likely to be replaced by useful ones, which are derived from hard real-life factors.

Technical development. The odds of success with developing something without good internal processes or clear business strategy are very grim.

Customers setting requirements. In the case study X was able to proceed to a point where it actually started communicating to pragmatist customers (X already had references which allowed this). In this situation the customers setting requirements were from a very mature level. Unfortunately X did not have its other levels matured sufficiently, which caused X not to reach any sales with the larger, pragmatist customers. Hence the indication here is, if there are immature development process and undeveloped competencies in the area of technical development, and additionally internal processes are immature and business strategy is not yet focused, approaching the major pragmatist customers of a given segment may not be remunerating for the software company. When tried, this type of approach may be remunerating for the airlines carrying the required business negotiators and experts.

8.4.1   The usability of the model

The significance of strategy and focus has been examined in this thesis both in the theoretical and in the case study chapters. As the discussion is all about software companies developing software, naturally the actual development issues are important also. While the companies and their upper management differ individually, it is impossible to identify a single cause why many of the software product companies fail to meet their targets. As there are multiple reasons for the management not to be able to create a competitive company, it is important to understand the critical focus areas in each phase of the company's development. It is equally important to piece together why a certain issue is critical at a given point of time and what kind of interrelationships and bearing these issues have with each other. The presented stage growth model from

tailoring to MOTS software product business allows the perception and situation of a software company in a holistic manner in a given point of time. Furthermore the model allows the management to see the probabilistic future of the young product company's requirements and the means to meet them, thus offering the possibility for the management to prepare for the situations proactively, instead of facing the situations as they come and trying to reactively work with them.

Another contribution of the model is its cross-organizational nature. The model allows the different units and their personnel to see the expected development and to understand what is their role in different points of time. Hence, adopting to change can be very much easier if one understands the reasons for it. As this understanding develops, it can act as a positive factor for planning the issues beforehand on a unit level, thus allowing for better motivation for the involved actors.

While the issue of focusing on the right issues is important, it can be equally important to understand what issues are not critical at a given time, and should not receive too strong focus immediately. For example building a strict, expensive quality system in a very early phase of the software company with a core group of developers may not make sense. This kind of action involves the use of significant amount of resources, which are taken from the crucial development and management work of the company. As the company needs to change its organization (and establish new roles) almost completely when the different units are established, the degree of usability of the first quality system can be questioned greatly, as can be the investment in it. The implicit assumption behind focusing on quality system (or any investment) is, that since resources are normally scarce, the allocation of them should be targeted to areas where the expected results become soon exploitable by the software company and remain exploitable for a relatively long period of time (failing in materializing the results soon after the equivalent investments have been made can hurt the company severely, since the competition should be assumed "doing the right things"). If the results of activities and corresponding investments can be accomplished simultaneously with the results being valuable and exploitable in the future also, the usage of the model can be evaluated of good understanding and quality. It is notable, that this model does not try

to reduce the need for or value of managerial work; the purpose is to try to offer tools for it.

8.4.2   Positioning the model

The software industry has some company-wide, holistic models that are generally used. One of these models is the Balanced Scorecard, which tries to offer a company-wide perspective. According to CorVu corporation (Demystifying Balanced Scorecard) "to achieve balance the methodology prescribes the strategic assessment of four perspectives: financial, internal, customer and learning and growth." By creating metrics on the performance of the company in these areas the company strives to improve its performance. Paul Arveson (Arveson, 1998) describes the need and the use of the metrics of the Balanced Scorecard in the following way:

"So the value of metrics is in their ability to provide a factual basis for defining:

- Strategic feedback to show the present status of the organization from many perspectives for decision makers

- Diagnostic feedback into various processes to guide improvements on a continuous basis

- Trends in performance over time as the metrics are tracked

- Feedback around the measurement methods themselves, and which metrics should be tracked

- Quantitative inputs to forecasting methods and models for decision support systems."

In general the Balanced Scorecard can be characterized as a holistic approach to a more mature company, with established processes, which can be incorporated with metrics mechanisms to further develop the already defined and established mechanisms. As this type of approach is important, it seems fair to assess, that the strong areas of the Balanced Scorecard are not in the startup companies trying to create new products, their relevant business models and the required internal processes. More mature companies

can benefit significantly from this type of monitoring and development work. As the stage growth model from tailoring to MOTS software product business helps to create these mature companies, the Balanced Scorecard can be seen as more focused on further developing them through process-built metrics and their analysis.

In general, it can be concluded many models have been presented focusing only to mature companies. Another group of focus is naturally the tailoring companies in general. The waterfall model can be seen as strongly originating from that field. Even the CMM process frameworks are criticized by Brownsword et al. (Brownsword et al. 2000, 50) to focus on tailoring, instead of the product domain. In historical perspective this is understandable, but also indicates today's need for stronger product-orientation as an emerged software process research area.

As the stage growth model has been a clearly identifiable contribution of this thesis, there are still other, more practical issues left to discuss. Unfortunately it is impossible to include the whole richness of product-related issues into any model. Many of these issues have been examined in the case study and in the earlier theoretical chapters. The next chapter discusses and further elaborates these issues.

9 PRACTICAL CONTRIBUTIONS

In this chapter some key issues are offered, these are identified as clear, practical contributions of this examination. The majority of the matters discussed have been investigated in the earlier chapters of this work. The purpose of this chapter is to offer them a worthy conclusion.

9.1 Validation of the weaknesses of the sales oriented approach

This approach clearly has multiple ramifications for the product company. Not selecting a segment is allowable for a relatively short period of time, when the product is in a very immature state and no one knows how or where it should be targeted. This type of situation however, should not last for a long period of time

In the worst scenario the sales orientation is so extensive, that the company does not select any clear segment at all to focus the company's software products. Instead the mainstream market is targeted immediately (as was the situation in the case study). This can easily lead to a situation where none of the studied levels of the company have actually developed to a maturity where their ability to execute is clearly above the required level (mainstream requirement level = pragmatists). In this type of situation the marketing and sales unit may not have enough time to develop their understanding of the situation and their competencies. The same applies to the software product development work and documentation. Another ramification of a very sales oriented approach is that while focusing on specific sales entities, the development of market oriented strategy and goals for the company may be vague or missing. This type of lack of direction can also hinder the development of the company's internal processes.

A clear suggestion is formed based on this study: if there does not seem to be any clear, feasible segment to work with, the option of bringing the whole product venture / company to an end should be seriously considered. If there exists business opportunities and the management is still not willing to choose any segment to work with, it should be explicitly documented why this is done and who exactly is able to carry the

responsibility for the type of behaviour. Possibly an outside consultant could be helpful in determining the reasons for the segmentation, or the lack of it.

9.2 Generic software products in Finnish and other international markets

It can be assumed that in Finland and in some other smaller markets in Europe the domestic market for MOTS software products is divided mostly between the companies existing within that particular country. Furthermore it can be assumed that this has been particularly true during 70's, 80's and in the first half of the 90's before the internalization effect caused by the internet and other international factors occurred. In this type of situation each domestic software segment may not have been under fierce competition. Instead some segments may have had only a small number of vendors, if there has been a domestic vendor / competitor for each segment to begin with. This can easily create a market situation where the success of a single, generically created software component has been possible in multiple segments of users. The lack of competition does not require every segment-specific feature to be encoded into the software product. As long as the major requirements are looked after, it may have been possible to carry out software product business successfully in multiple market segments in this manner. Supporting these assumptions is the following statement (Autere et al. 1999, 39): *"The level of productization required to directly sell software in Finland is not sufficient for the purposes of an international sales channel."* They identify multiple issues that are lacking from the required international level.

According to (Autere et al. 1999, 38) there is a segmentation problem with the Finnish software companies who have gotten used to a very little need for segmentation in the domestic market. The international software market is however somewhat different in market logic. Regarding software products in the international market they note: "For a small software company creating competitive advantage, known reputation and credibility can be generally accomplished only through the selection of a very narrow segment. Even in that segment they must focus on one or two features producing key competitive advantage" (Ibid). Furthermore they state that for companies operating in Finnish markets the need for segmenting and focusing requires a whole new learning

process: "The company must select only one from their ten products and ten customer segments and even narrow the selected one" (Ibid). The indication here is relatively clear, the non-segmenting assumption, which may be valid in Finnish and some other European domestic markets, is not valid when entering the international software product business area. When this type of assumption is used in the international market, as was done by the management of X, the results are immediately in jeopardy. Using this assumption can at first work relatively fine when entering a new, innovative software product market in its early stages. At this time there is very little competition and the customers' requirement level is typically low due to the new nature of the whole phenomenon. It is in the next phase of development when many of the problems begin to show. After the market for the product becomes more widely known internationally, there is always plenty of competition, assuming anything else can be characterized as "courageous and fearless." As the international market is formed, it is more often than not relatively well developed, thus containing many strictly focused, segmentation oriented competitors. The software product company having "generic code to be used in multiple purposes" finds itself in trouble trying to answer to all the large, high level and other, more detailed requirements from all the possible different segments. In mathematical terms this means that if the company's resources equal 1 and those resources allow the advancement speed in a given segment to be also 1, these resources and speed can then be divided by the number of segments selected (assuming there is similar amount of resources allocated to each segment). If there are three segments selected, the company's ability to perform in each segment is then 1/3 of the possible, more focused option the competition can be assumed to exercise.

As the product market in question develops, this level of ability to execute guarantees the competitors to be moving in greater speed to introduce new features, concepts, business models etc. into the market. This leaves very little possibility for the "generic", "multi-segment" or "multi-product" company to succeed. Regarding the number of the products, it is notable that if the products absolutely support each other in a single segment and can be sold through the exact same sales channel arrangements; it is difficult to determine that the number of MOTS products has to be precisely 1. In the COTS product domain a small software product unit / company having more than one

products may find it more difficult to support them in the market, especially if those products differ in their target segments extensively.

9.3   Operating tailoring and product business in a single company

Both the presented theory and the case study research strongly indicate, that although this approach can seem to be a feasible arrangement, which minimizes risks, after a closer examination this is not commonly the case. This issue can also be seen to have a relation with selecting the company and product focus. The main flaw of this approach seems to be found from the managerial area. This can be seen true especially for small and medium-sized companies. Both areas of business require their own managerial knowledge to be gathered and processed, as is discussed in chapter 2. This knowledge should then be reflected in the different strategy areas of the software company. It is very probable, that the software company's management does not have the resources to process (and implement accordingly) the two different domains of knowledge in their respective, transforming, competitive markets. This acts as a future negative factor for the company to be able to organically grow significantly in either of the sectors.

One option is to form two different units (or companies), which can have separate strategies. This was partially tried in the studied company X. Even in this approach the limitations of sufficient, experienced management are difficult to overcome. In the gloomiest scenario neither of the units have sufficient resources and talent in the management causing both units to suffer from this. One feasible option is to reach a decision which business is seen as the core for the future, and sell the other form of business to fund the other one. For a large corporation maintaining the two units can be an option, provided both of the units are staffed sufficiently. In that case some synergy advantages can also be realized, if both of the units are able to function profitably as standalone units, and at the same time they are able to assist each other.

9.4   Company's communication strategy in MOTS product business

Communication can be seen as a very important area for any company and its customers. The special quality of MOTS companies is that even when they are selling products, they yet need to sell and often participate in tailoring tasks implemented at the customer's premises. Compared to the COTS business this can be seen as an extra complexity carrying out the business. If the software company's interest area extends only to the sales channel in COTS business, this type of software company can focus its communication mechanisms greatly only to that area. Naturally there is the Internet to provide a communication channel even further, but the core business-related communication mechanisms typically need to be established only as far as the sales channels are. On the contrary the MOTS company needs to establish communication mechanisms all the way to the customers, especially if one considers the smaller companies who may not be able to persuade the sales channels to learn everything regarding their product, this is very likely to be the case.

More complexity is added to the situation when not all sales channels are willing to allow the free communication between the software company and the customer. From their perspective owning the customer is important and they do not want to be replaced by any direct relationship between the customer and the software company. This is only natural, however this means that on the other hand there are benefits to be gained by the software company of such interaction and on the other hand there is a clear motivation to at least limit the communication, if not prevent it. For the communication to be fluent the sales channel can be presented a good model supported with a legal contract offering both parties the access and security they need to carry out their businesses. This type of a model should be designed, presented and accepted prior to the beginning of actual cooperation. After the cooperation has started and the sales channel has already adopted a very proprietary relationship with the customer, it is very difficult to suggest the rules to be changed only because the software company benefits from it.

To be able to generate added value for the software company, the interaction mechanisms, roles and responsibilities and feedback mechanisms between the software company, sales channel and the customer are suggested to be carefully designed. Since

the task to be accomplished can be seen even more complex than in the case of COTS companies, this task can prove to be a very difficult one. The studied company X clearly was not able to create a communication strategy / model, which would have ensured the required information for every participant in the form of a fluent process.

Based on this study it is identified also as a risk to let the sales unit to be responsible for the communication alone. There seems to be the risk of not gaining all the required valuable product feedback. This can be true at least in a situation where the software company has relatively unestablished processes and the sales task is carried out relatively autonomically by the sales unit / personnel. The customers and the consideration of their (often technical) requirements are essential to develop a competitive software product. In this type of situation the deep understanding what the customer's unit in charge of MOTS company's product thinks and having close personal relationships with customer's key staff is important. As the problem field is described here, designing a good model for this type of communication strategy is beyond the scope of this thesis. However, it can be recommended to be a part of future research.

10 CONCLUSIONS

This study has examined software product business and MOTS software product business in particular. The results show a large emphasis on the management, strategy and focus. While the tailoring-oriented software company is able to succeed with a relatively simple strategy of mastering the technologies in a certain area of business, the product-focused company is required to do much more strategy and managerial work. This work is related to company focus, product positioning, pricing, communication with customers and partners, logistics and service strategies.

To be able to accomplish the successful MOTS orientation successfully the software product company requires additional resources, often scarce in the domestic market. A large portion of these newly required resources are managerial and require international product experience. The study did not show the lack of technical personnel, although the testing of software products can be viewed as a partially technical, immature area of young product companies.

The internal implementation of transformation is started, when the newly hired (or repositioned) staff is allocated to fulfill new positions in the company, thus requiring the internal processes and former allocation of tasks to be adjusted also. To carry out this level of change requires plenty of managerial attention and planning. The developers and other staff of the software company are better motivated to carry out their tasks, if the transformation and its underlying rationale are explicitly planned. The stage growth model developed during this examination is a tool for the management, as well as for the other staff of the company to enable the direction and correct focus to be developed for the company in each phase of the company's development. If understood and used correctly, it can serve as a planning base to allow the software company to move from reactive mode of functioning toward a more proactive, methodical way of carrying out software business.

Further results are shown in the form of recommendations regarding sales oriented strategy and generic software products in the international software market. These have been described with the communication strategy in the (subchapters of) chapter 9. In the

light of this thesis these two approaches are not recommendable and should be avoided. Instead, the selection of strategy and relevant product focus should be formed relatively early. The ability of the small or medium sized software product company to compete is enhanced greatly, if it can compete only in one segment of products and customers internationally. The findings of this study also show, that domestic software product experiences can even be harmful in the international environment, especially if they are implemented directly to the international environment. This again emphasizes the need for internationally experienced upper management.

Overall, the results of this examination suggest that it is beneficial to develop the MOTS software product company holistically. This includes the selection of target customers in relation with the maturity level of the software product(s), current resources and competencies of the software company. Allowing the target decisions to be affected by any kind of unrealistic enthusiasm can be destructive for the materialized results for the software company. This is caused by the misallocated focus and tasks of the company to a segment, where the is no match between the maturity of the product, company's internal processes and competencies, and the requirement level of the given targeted customer group. In here the company's true ability to execute the required tasks plays a key role in materializing the expected economic results. A further, related suggestion is that sufficient time to practice should be allowed for the whole company to be able to respond to and exceed the higher requirements presented by the pragmatist customers.

The exact research problem of this thesis was described by the definition of three sub-problems: 1. how should the product be developed in different phases, 2. how do the internal processes develop during the period of productization and what competencies are required within the company during this same period, and finally, 3. what should be the marketing and business strategy focus areas during this same period? The literature study, case and the presented model offer understanding for these issues. To describe and answer the issue of the product's development in different stages the NPD process, the rationale behind product focus, and the rationale behind feature selection were examined. This area was interconnected to the development of the company strategy in different areas and the development of the software company's competencies and processes. Goeffrey Moore's work relating to marketing strategy received special

interest as a key factor for success. As a relatively holistic, yet adequately detailed answers and descriptions to the research questions were formed during this study, it is notable that the whole MOTS software product domain is still in its teens, as is this specific research area. Reflecting to the level of maturity of the research domain, the holistic quality and level of the results of this research seem justifiable. At the same time it must be noted, that as the MOTS software research area matures in the future, there is a need to describe many of the discussed areas in much greater detail and produce more validation than accomplished in this study. For an immature research area the selected, holistic research focus seems justifiable.

Future research topics formed from the area of this research are as follows: To further investigate how does the presented stage growth model behave in the COTS, start-up, corporate venturing and venture capital environments, and how could the model be developed to respond to the special qualities of those environments. The second area of future research is the relationship between domain engineering and application engineering, and how these should be combined in different stages of the software product's maturity. One research topic stated earlier in this thesis was the design of a model for communication strategy for young, small and medium-sized MOTS software companies. Furthermore a detailed description of the technical development level, requirements management with applied COTS RM software, version control and version generation, multi-interface software product testing are all areas for future research.

Based on this study it seems (FIGURE 6,7,), (Brownsword et al. 2000, 50), that much of the field of academic information systems, information science, software engineering and systems development is actually more related to tailoring and related processes, and many of the models and suggested processes are not directly applicable to the software product domain. Many of the widely accepted software development methods and process models seem to have their roots in and be more focused on tailoring software solutions. It must be stated, that these two domains have something in common, but it can be questioned whether this similarity is sufficient to claim that the tailoring-rooted models are applicable as such in the product domain. This leaves the product-specific requirements partially unanswered by many of the commonly known models and

suggested processes. This suggests a vast area of examination of the product domain and the creation of similar company- and unit-level processes and approaches with similar fidelity of translation as has been implemented by the academia in the past relating to the tailoring area. Evidently, we are not discussing of two, identical entities creating software.

As the partially immature state of software product related processes and software product companies in Europe can be viewed as the status quo currently, the challenges of creating successful B to B, MOTS products are waiting to be answered by many European software companies.

11 REFERENCES

Arveson P. 1998 [online]. What is the balanced scorecard. Available in the www-format: <URL: http://www.balancedscorecard.org/basics/bsc1.html>. The Balanced Scorecard Institute, (Read 2.5.2002).

Athaide G.,A., Meyers P.,W., Wilemon D.,L. 1996. Seller-Buyer Interactions During the Commercialization of Technological Process Innovations. Journal of Product Innovation Management, No. 13, 406-421.

Autere J., Lamberg I., Tarjanne A. 1999. Ohjelmistotuotteilla kansainväliseen menestykseen. Helsinki, National Technology Agency in Finland (TEKES).

Brownsword L., Oberndorf T., Sledge C.A. 2000. Developing New Processes for COTS-Based Systems. IEEE Software, No. 4, 48-55.

Carmel E. 1994. Time-to-completion in software package startups. Collaboration Technology Organizational Systems and Technology Proceedings of the Twenty-Seventh Annual Hawaii International Conference on System Sciences, HI, 498-507.

Cooper R.G. 1996. Overhauling the New Product Process. Industrial Marketing Management, No. 6, 465-482.

Cooper R.G. 1999. FROM EXPERIENCE. The Invisible Success Factors in Product Innovation. Journal of Product Innovation Management, No. 16, 115-133.

Cooper R.G. 2001. Winning at New Products. Accelerating the Process from Idea to Launch. Third edition. Massachusetts, Perseus Publishing.

Cusumano M.A., Yoffie D.B. 1998. Competing on Internet time: lessons learned from Netscape and its battle with Microsoft. New York, The Free Press.

Cusumano M., Yoffie D. 1999. Software Development on Internet Time. Computer, October, 60-69.

Davis R.E. 1996. Market analysis and segmentation issues for new consumer products in The PDMA Handbook of New Product Development. New York, John Wiley & Sons,Inc.

Demystifying Balanced Scorecard [online]. Available in the PDF-format: URL:http://www.corvu.com/library/whitepapers/index.html. Corvu corporation. (Read 2.5.2002).

Eisenhardt K. 1989. Building Theories from Case Study Research. Academy of Management Review 14 (4), 532-550.

Hoch D.J., Roeding C.R., Purkert G., Lindner S.K., Müller R. 1999. Secrets of software success. Management Insights from 100 Software Firms around the World. Boston, Harvard Business School Press.

Käkölä T. 2001. Organizational process, and Information System Designs for Facilitating Process Exploration and Exploitation in Corporate Software Ventures. Unpublished paper. University of Jyväskylä, Jyväskylä.

Käkölä T. 2002 [online]. Software Business Models and Contexts for Software Innovation: Key Areas for Software Business Research. Available in the www-format: <URL: http://www.exba.net/default_NS.asp?gc=261&g=k&c=2>. Center for excellence Finland. (Read 14.5.2002).

March S., Smith G. 1995. Design and natural science research on information technology. Decision support systems, 15, 251-266.

McHugh P. 1999. Making it Big in Software – a guide to success for software vendors with growth ambitions. NW, Rubic Publishing.

Miettinen R. 1998. Methodological issues of studying innovation-related networks. Working papers n:o 4, Technical Research Centre of Finland – Group for Technology Studies.

Moore G.A. 1995. Inside the Tornado. New York, HarperCollins Publishers Inc.

Moore G.A. 1998. Crossing the Chasm. Second edition. Oxford, Capstone Publishing limited.

Ottum B.D., Moore W.L. 1997. The Role of Market Information in New Product Success / Failure. Journal of Product Innovation Management, No. 14, 258-273.

Parvinen P. 2001. SoftaPro product manager training 8.5.2001. Espoo, Helsinki University of Technology.

Rajala R., Rossi M., Tuunainen V., Korri S. 2001. Software Business Models, A Framework for Anlyzing Software Industry. Helsinki, National Technology Agency in Finland (TEKES).

Roberts E.B. 1990. Evolving Toward Product and Market-Orientation: The Early Years of Technology-Based Firms. Journal of Product Innovation Management, No. 7, 274-287.

Rosenau M.D., Griffin A., Castellion G.A., Anschuetz N.F. 1996. The PDMA Handbook of New Product Development. New York, John Wiley & Sons, Inc.

Räsänen H. 2001. YRI C66 High-Tech Enterpreurship course material. University of Jyväskylä, Jyväskylä.

Salo A., Käkölä T.K. 2002. Groupware Support for Requirements Management in New Product Development. Unpublished paper accepted for Journal of Organizational Computing and Electronic Commerce. University of Jyväskylä, Jyväskylä.

SoftaPro Product Manager Training 2001. Discussions with other product manager attendants. Helsinki University of Technology, Helsinki.

Souder Wm.E., Buisson D., Garret T. 1997. Success Through Customer-Driven New Product Development: A Comparison of U.S. and New Zealand Small Entrepreneurial High Technology Firms. Journal of Product Innovation Management, No. 14, 459-472.

Trott P. 1998. Innovation Management & New Product Development. Essex, Pearson education limited.

Tähtinen J. 1999. The Existence and the Dissolution of a Business Relationship in Tailored Software Business: The Theoretical Framework. University of Oulu, Oulu.

Warsta J. 2001. Contracting in software business. Doctoral dissertation. University of Oulu, Oulu.

Äijö T. 2001. Suomalaisyritys kansainvälistyy: strategiat, vaihtoehdot ja suunnittelu. Helsinki, FINTRA.