

Università degli Studi di Modena e Reggio Emilia

**DOTTORATO DI RICERCA IN
INGEGNERIA DELL'INNOVAZIONE
INDUSTRIALE**

Ciclo XXVIII

**Systems Engineering to support Service
Product Development**

Candidato: **Gaetano Cutrona**

Direttore della Scuola:
Prof. Mauro Dell'Amico

Relatore:
Prof. Cesare Fantuzzi

Esame finale anno 2016

Copyright ©2016 by Gaetano Cutrona.

Reggio Emilia, Italy, March 2016.

Abstract

We developed a process and related methodologies to support the growth of the Service Products Business in a packaging company which core business was selling packaging machines and packaging materials. The presented work focuses on establishing the framework and standardise the available best practices in order to build, around the company core business, additional possibilities to expand the company scope and provide additional services and products to increase customer satisfaction. Competition in the Industrial market is growing and growing and customers physiologically become more and more exigent and competent. A big Company, thus, is forced to provide full solutions in own portfolios if it would like to cover customer needs and support products in the entire life cycle phases. Services are becoming quite important today because they are seen as part of the product and often they are determining the choice of a product rather than another one. The process and methodology will support the shift from pure manufacturer to product-service deliverer for companies willing to change the rules of competition. We started identifying the problem and create a model of the company involving strategy, business model, value chain and products lifecycles. The classic V-Model was identified as framework for the process workflow but according to some stakeholder requirements we wanted to introduce also some AGILE methodologies in specific phases of service development process. The Systems Engineering Principles (holistic view, structured approach and complexity management) were the drivers of all the thesis activities. During the work we took into consideration the human factors and the organizational setup of the company in order to quickly enable the new products introduction in the R&D department. We used a pilot project to verify and validate the methodologies in a real case study. The results of the activity consist of a fully integrated gate based process with methodologies and tools that support service products developments.

Sommario

Abbiamo sviluppato un processo e le relative metodologie per sostenere la crescita de Business dei Service Product in una società di Food Packaging il cui core business era basato in macchine e materiali per l'impacchettamento. Il lavoro presentato si concentra sulla creazione del framework e la standardizzazione delle migliori pratiche disponibili, al fine di costruire, intorno al core business aziendale, nuove possibilità per ampliare il mercato dell'azienda e fornire servizi e prodotti aggiuntivi per aumentare la soddisfazione del cliente. La concorrenza nel mercato industriale è in continua crescita e i clienti fisiologicamente diventano sempre più esigenti e competenti. Una grande società, in tal modo, è costretta a fornire soluzioni complete nel proprio portafoglio prodotti per coprire le esigenze dei propri clienti e supportare i propri prodotti in tutte le fasi del loro ciclo di vita. Al giorno d'oggi i servizi stanno diventando molto importante perché sono visti, sempre più, come parte del prodotto stesso e spesso sono determinanti per la scelta di un prodotto piuttosto che un altro. Il processo e la metodologia sviluppati supporteranno il passaggio dal puro produttore manifatturiero alla fornitura di prodotti e servizi per quelle aziende che vogliono cambiare le regole della concorrenza. Abbiamo iniziato l'attività' identificando il problema e creando un modello dell'azienda considerandone: strategia, modello di business, catena del valore e ciclo di vita dei prodotti. Il classico V-Model è stato identificato come framework di riferimento per il flusso del processo, ma secondo alcune esigenze degli stakeholder abbiamo voluto introdurre anche alcune metodologie AGILE in alcune fasi specifiche del processo di sviluppo. I principi del Systems Engineering (visione olistica, approccio strutturato e di gestione della complessità) sono stati i piloti di tutte le attività di tesi. Durante il lavoro abbiamo preso in considerazione i fattori umani e la configurazione organizzativa dell'azienda, al fine di consentire rapidamente la nuova introduzione del service product development nel reparto R&D. Abbiamo usato un progetto pilota per verificare e convalidare le metodologie in un caso di studio reale. I risultati dell'attività sono

costituiti da un processo “gate based” completamente integrato con metodologie e strumenti che supportano lo sviluppo dei service product.

Acknowledgements

This thesis is the result of several years of work, and I really need to thank who made it possible. I would like to express my deepest thanks to my supervisor, Prof. Cesare Fantuzzi for giving me this opportunity and for his help, teachings and support during these years. Many thanks also to my colleague and friend Andrea Margini that supported me a lot in this journey. Our days in the office environment were full of humor and entertainment and this allowed me to always work with the right soul.

I really need to thank my son Ignazio and my daughter Desiree for their smiles and energy during these years. Their presence is an essential constant, inside the ups and down of my life. Thanks for everything and especially for your patience.

Thanks a lot also to my new baby, Eliana, that being born a few days before the completion of this work, has enlightened me to a new shining light. And, of course, my greatest thanks go to Karoline. Thanks for being there, always.

Contents

1	Introduction	1
1.1	Product/Service-System Strategy	2
1.2	Systems Engineering	3
1.3	Service Product Creation Process Design	7
1.4	Contribution and thesis outline	12
2	Core Process Map	15
2.1	introduction	15
2.1.1	Outline	18
2.2	Process Mapping Methodology	18
2.3	Resulting Process Map	20
2.4	Discussion	25
3	The Service Development Process	27
3.1	introduction	27
3.1.1	Outline	30
3.2	The Service Development Process	30
3.2.1	Discover phase	32
3.2.2	Define phase	33
3.2.3	Develop phase	34
3.2.4	Deploy phase	35
3.3	The Value Chain vs Product Life Cycle Approach	37
3.3.1	Methodology Description	38
3.4	Product Architecture Management - an approach to Product Life Cycle	41
3.4.1	Background	42
3.4.2	Methodology Description	43
3.5	Discussion	49

4	Case Study and process validation	51
4.1	Introduction	51
4.1.1	Outline	53
4.2	Case Study	53
4.2.1	Requirements collection and analysis	54
4.2.2	Architecture design activities	54
4.2.3	Verification Activity	62
4.3	Comparison with classical design activities	64
4.4	Discussion	65
5	Concluding remarks	67
	Bibliography	69

Chapter 1

Introduction

A steady pressure on manufacturing companies is one of the consequences of the increasing global competition in our post-modern society in the era of mass consumption over the past decade. The adoption of concurrent engineering processes [6] has enabled companies to differentiate their products with respect to the three key competitive aspects: Cost, Quality and Time (Figure 1.1).



Figure 1.1: The three key competitive aspects of a product

However competing has proven difficult for manufacturing organizations in developed countries, as low labor cost countries have increased their market freedom and thereafter their industrial capacities. Companies are now looking for new business strategies and market differentiators beyond cost, quality and time. The manufacturing organizations are currently concerned about the future of their business. Especially the ones that have based their business on the production and sale of technologically mature products. In this specific case, the traditional business model of differentiating the product respect price, functionality and design led to concerns related to the future profitability of the products. The simple strategy of decreasing product costs has been proved unsuccessful so many companies are now focusing on increasing their understanding on how customers perceive

the value of their products. Given these trends, very little research has been done in the development activities of manufacturing companies that are in the process of redefining the value of their offering [43]. In this thesis we intend to investigate and describe the evolution of the process and related methodologies for a company that produces and sells food packaging equipment. The work shows how this global manufacturing company is addressing the problem in a development project by evolving its processes from selling packaging equipment to sell full packaging services. In few words we show how the change from producing packaging equipment to the production of full service/system solution may enhance the value of the company products.

1.1 Product/Service-System Strategy

The followed approach of re-defining the company's offering value is called Product/Service-System (PSS) [18] and aims to the adoption of a product/service business oriented strategy. The innovation of strategies like the Product/Service business one is characterized by changing the perspective from the product itself to activities associated to the use of the product [2]. In general companies adopting such strategies consider that the value is not embedded in the physical product but is created by supporting customer's activities related to the use of the product. More value for the customer is then created if the entire life cycle of the product is considered (Figure 1.2).

With this approach, companies could rationalize, enhance and reuse their products and services by considering the life cycle phases. On the other side, Customers are not to be concerned with the responsibility to learn how to use, maintain or dispose products but simply take advantage from the effects of use. Basically The PSS approach pushes the companies to integrate products (physical artifacts) with services (capabilities, knowledges, experience and activities) to create more valuable offerings. In order to accomplish this task the company must reconsider the network of actors [32] that usually operate within the business with the additional point of view of customer's activities and product life cycle stages. The adoption of a PSS strategy, then, is characterized by the following opportunity parameters [43]:

- Re-Invention of Core Business Areas
- Increasing of competitive edge (customer lock-on)
- Improved control over the Product Life-Cycle
- Greater understanding of the product's use and nature

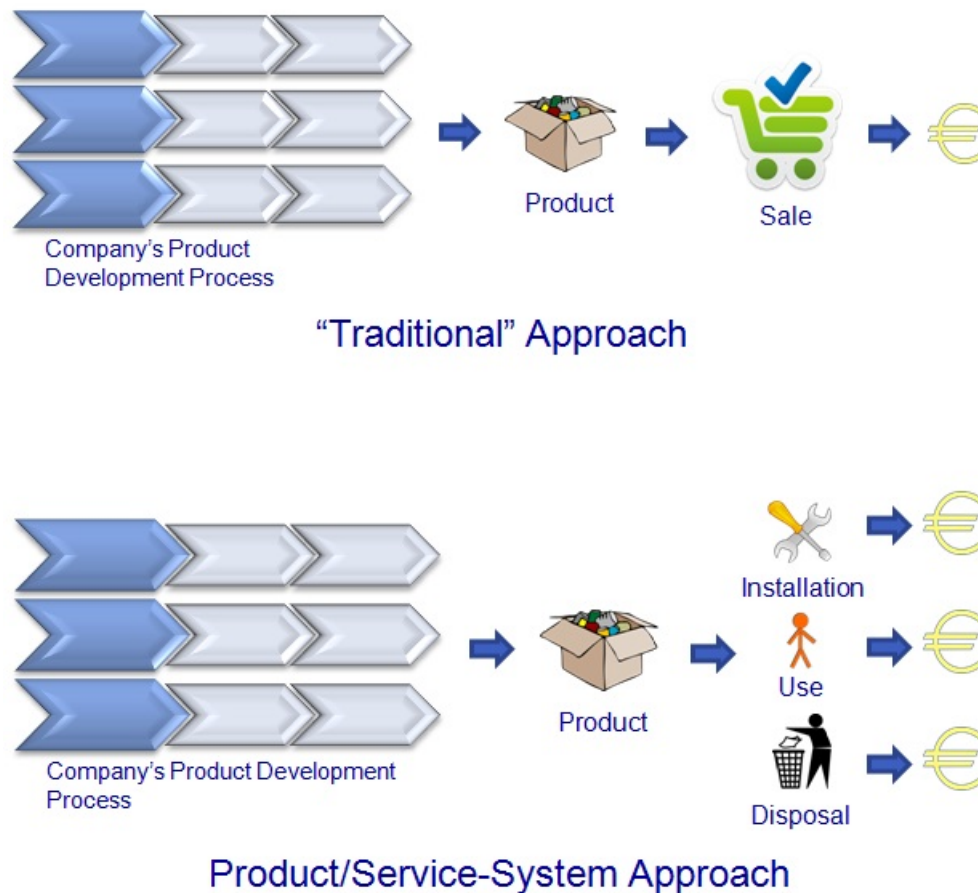


Figure 1.2: Differences between the "Traditional" Approach (value creation finishes with the product sale) and PSS approach (value is created in each stage of the product life cycle).

- Sustainability predisposition

Traditional manufacturing companies perform product and service development activities in different parts of the organization. The PSS approach embodies the cooperation and integration of developments activities since it switches from a "physical system" thinking to a "functional system" thinking.

1.2 Systems Engineering

Systems Engineering (SE) is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem:

- Operations
- Cost & Schedule
- Performance
- Test
- Manufacturing
- Training & Support
- Disposal

Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs [22]. The concepts of "System" can be generally traced back to early western philosophy and later to science, the most common to engineers is attributed to Ludwig von Bertalanffy (1950, 1968). He explained a system as a set of interacting or interdependent components (sub systems) forming an integrated whole. A system boundary separates the system of interest from its context (Figure 1.3).

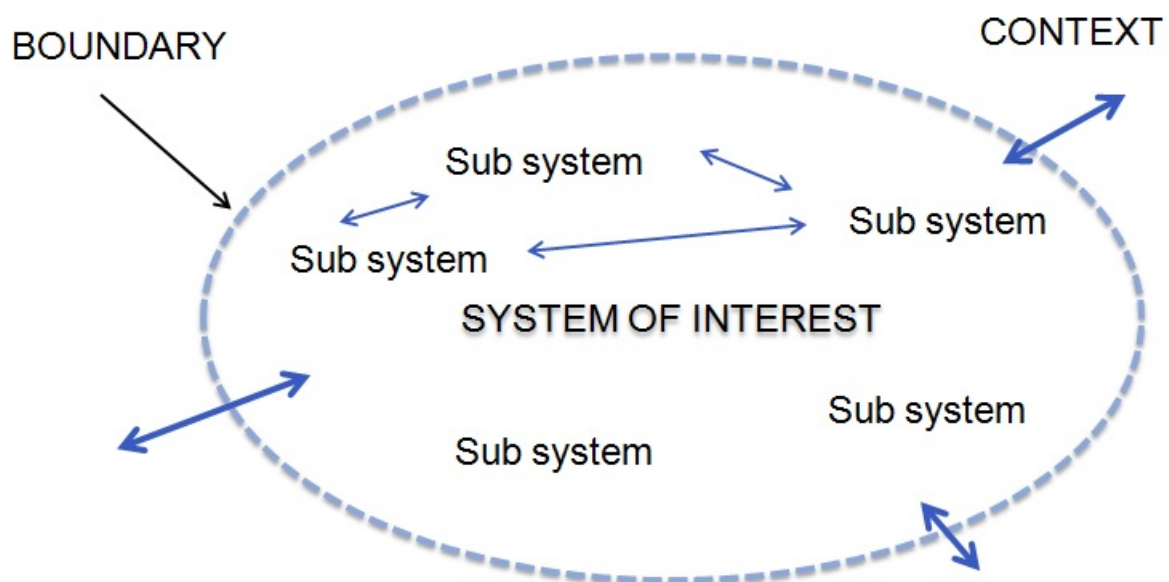


Figure 1.3: Definition of System: Set of interacting parts separated from its context by the boundary.

The above concept of a system is further explained and detailed by INCOSE that together with ISO/IEC/IEEE created the following definitions [22][26]:

- "... *an integrated set of elements, subsystems, or assemblies that accomplish a defined objective. These elements include products (hardware, software, firmware), processes, people, information, techniques, facilities, services and other support elements.*"(INCOSE).
- "... *combination of interacting elements organized to achieve one or more stated purposes.*"(ISO/IEC/IEEE 15288).

From these definitions we can easily understand that by viewing the system from an external perspective, there are elements that specifically do not belong to the system but do interact with the system. This view is called *Operating environment or context* and it could include the users of the system. The *system boundary* separates the system from its context and it defines what belongs to the system and what does not. A system is described also by indicating its *functionalities* (what the system provides in terms of interactions) to the operating environment. A functionality is not only depending on how the single system element interacts with the context but it is also the result of how the interaction is influenced by the "organisation" of the system elements. This concept puts the basis for the following definition of system architecture[25]:

- "*the fundamental concepts or properties of a system in its environment embodied in its elements, relationship and in the principles of its design and evolution.*"

This definition considers both the internal and external views of the system and shares the concepts from the definition of a system. The modern origins of Systems Engineering can be traced to 1930s followed by other programs and standard definitions (Figure 1.4) [22].

The main benefit using SE approach is the effectiveness of the methodology to manage complexity, change and risks. SE should be used early in the development phases and continued to be applied during each life cycle phase of the product [22]. The need of SE can be easily understood by looking at the generic relationship between the life cycle phases and system costs. Next figure 1.5 shows an example of the trends for the cost committed and cost incurred curves in a typical product life cycle. As can be seen, about 80% of the cost of the system is committed by the end of Design and Integration, while only about 20% of the actual cost for the system has been spent. Obviously, mistakes made in the front-end of the system life cycle can have substantially big impacts on the total cost of the system and its success with the users and bill payers [6].

1969	Mil-Std 499
1979	Army Field Manual 770-78
1994	Perry Memorandum urges military contractors to adopt commercial practices. EIA 632 IS (interim standard) and IEEE 1220 (trial version) issued instead of Mil-Std 499B
1998	EIA 632 released
1999	IEEE 1220 released
2002	ISO/IEC 15288 released, adopted by IEEE in 2003
2012	<i>Guide to the Systems Engineering Body of Knowledge (SEBoK)</i> released

1969	Mil-Std 499
1979	Army Field Manual 770-78
1994	Perry Memorandum urges military contractors to adopt commercial practices. EIA 632 IS (interim standard) and IEEE 1220 (trial version) issued instead of Mil-Std 499B
1998	EIA 632 released
1999	IEEE 1220 released
2002	ISO/IEC 15288 released, adopted by IEEE in 2003
2012	<i>Guide to the Systems Engineering Body of Knowledge (SEBoK)</i> released

Figure 1.4: important dates of SE as discipline and standards.

We can identify, then, three main principles characterizing the Systems Engineering approach:

- A holistic view of:
 - The system, its subsystems and interfaces
 - The involved technology disciplines and their interactions
 - The complete system life-cycle
- A structured approach to ensure things are done right from the start:
 - A uniform workflow from Requirements definition to System validation
 - Define what is to be done before defining how it is to be done
 - Work iteratively to ensure learning from experiences
- Manage complexity by:
 - Trading off conflicting interests to ensure an optimized design
 - Addressing risks as early as possible where the cost impacts are lowest
 - Focusing on system interfaces during definition to reduce time during integration

Systems Engineering receives a great attention in current literature. In particular, the SE process for structuring work flow and activities related to the development of complex mechatronic artifacts have been studied and presented by several contributors. For example the VDI (the association of German Engineers) published guidelines [49], [48],

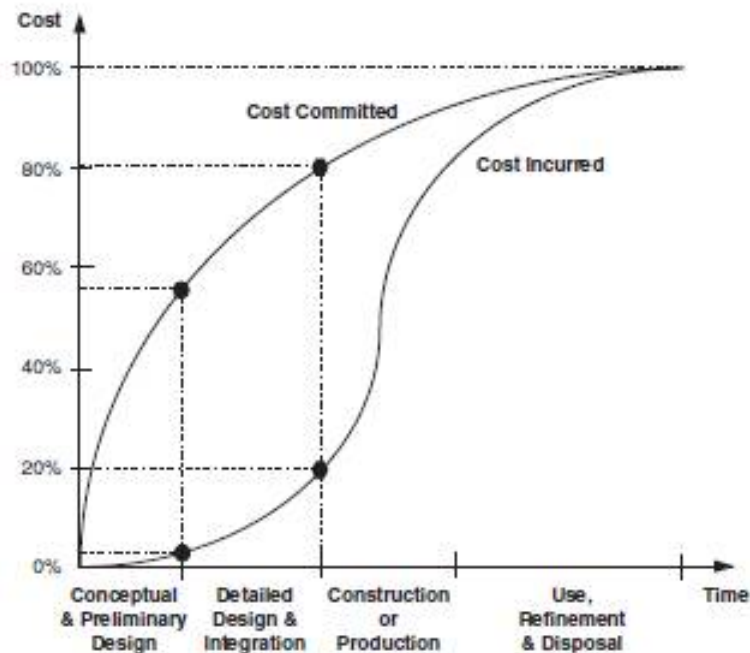


Figure 1.5: Cost commitment and incursion in the Product Lifecycle [21][6]

dealing with the development of a modern mechatronic product in its entirety. In this way it creates an essential basis for the communication and cooperation of experts in the disciplines involved. Recently, the Guide to the Systems Engineering Body of Knowledge (SEBoK) has been published, [38]. It provides a compendium of the key knowledge sources and references of systems engineering, organized and explained to assist a wide variety of users. It is a living document, accepting community input continuously, and regularly refreshed and updated. The results of this thesis have been obtained during the "Service Product Creation Process Design" project, addressing the implementation of SE in a packaging company while redesigning the development process to support new Service Products developments.

1.3 Service Product Creation Process Design

The main goal of the Service Product Creation Process Design project (SPD) is to develop and release a development process to support the service product development for a multinational packaging company leader in the manufacturing of equipment and materials market for food packaging. As described in [11], The global consumer packaging market is valued at approximately US\$400b and an estimated US\$500b if industrial end markets are included. The Brazil, Russia, India and China (BRIC) markets comprise approximately

30% of global demand, increasing as their economies further develop. Packaging sales in the emerging markets are expected to continue to show strong growth as both increased consumption and demand for consumer goods drives the need for more sophisticated packaging, due to a growing middle class. The sector includes five main types of packaging. Paper and board (including paper bags and cartons) is the largest consumer packaging category with a ~34% share of the total packaging market. Rigid plastics (tubs, pots and jars etc.) is the second-largest packaging category with a ~27% share and is one of the faster growing categories, forecast to grow above real GDP (~4% per year) (Figure 1.6). The macroeconomic environment has been challenging for the packaging industry in recent years, given pressures on consumer spending and their exposure to fast moving consumer good (FMCG) producers. The combination of Eurozone economic uncertainty and raw material and energy price inflation has also had a negative impact on packaging producers. Growth in emerging markets has been both a threat and an opportunity. These factors are some of the obstacles to being successful as a packaging producer.

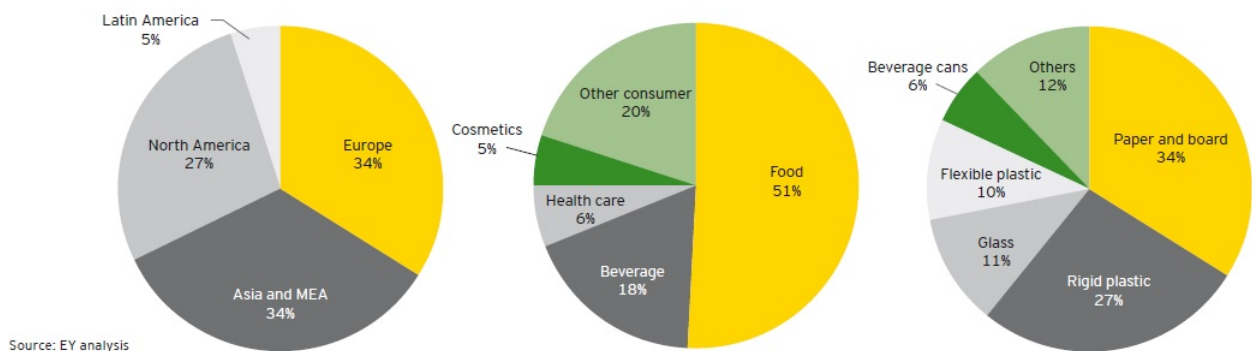


Figure 1.6: Global Packaging by geography, end market & type

The company, that started the SPD project, has a tradition of over 60 years of business in the food processing and packaging market. Its products are divided into several different categories:

- **Processing Solutions:** equipments and solutions to treat and process raw food to enable customer's brand products.
- **Packaging Solutions:** equipment and solutions to pack food into packages by keeping food characteristics and prevent external contaminations or alterations.
- **Distribution Solutions:** equipments and solutions to create distributable units of packed food to safely enable operations and food transportations to customer's

warehouses, retailers and consumers.

- **Service Products:** cover every aspect of customer's food production, from daily routines to business insights. Service solutions improve performance, optimize costs and ensure food safety throughout the life cycle of customer's operation.

the organization is divided in three main divisions to perform its operations:

- **Marketing:** responsible for customer relations, communications and marketing activities.
- **Production:** it is in charge for R&D developments, technical service activities and product life cycle management.
- **Supply Chain:** responsible for suppliers management, spare parts management and sales support.

A project organizational setup is characterizing its development operations. This means that the major part of the development activities are gathered thru the conduction of a project that at its completion will deliver the product. Resources belonging to the three above divisions participate in order to deliver the project goals (Figure 1.7)

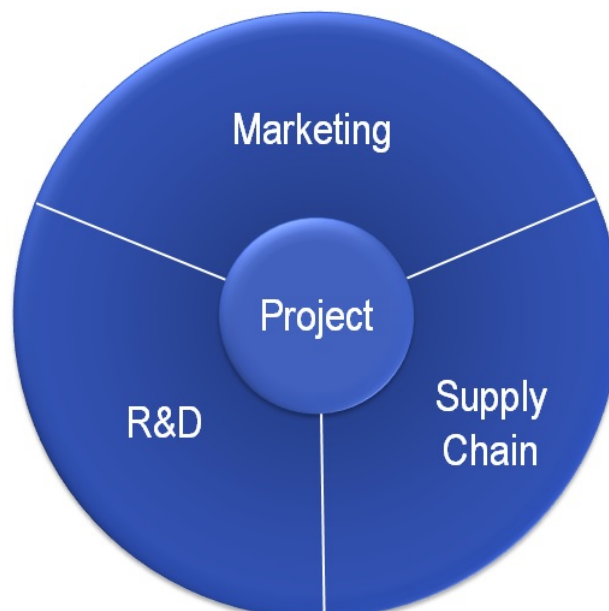


Figure 1.7: Resources from each division of the Company participate to each product development project

The implemented Product Development (PD) process appeared as a phase based process with 5 phases divided by 6 Gate Reviews (Figure 1.9). Per each of them the project governance decides if to continue, adjust or stop the project.

During each phase there are some recommended DRMs (design review meetings) at system and subsystem level which aim to recommend activities to the project or to assess its technical maturity.

To represent the project team organization and responsibilities, the workflow has been organized in lanes (e.g. project & product management, system engineering, subsystems design, etc.). Each PD project is recommended to follow the above mentioned process. Before the project startup a project governance is created by the company management team and a Project Manager is appointed as project completion and delivery responsible. A core team (Figure 1.8) is created by the newly created governance together with the project manager in order to provide the right competences and responsibilities to fulfill the project scope. Each core team member (including the project manager) is responsible of the provision of the deliverables. In general the core team is composed by a project manager (responsible for both the entire project progress and schedule and marketing activities), a systems engineer (responsible for the system technical solution), a development core team member (in charge to fulfill the subsystems requirements) and finally a supply chain responsible (to provide the supply chain support). Each core team member is supported by an extended team composed by people coming from different line organizations in order to fill eventual competence gaps [23].

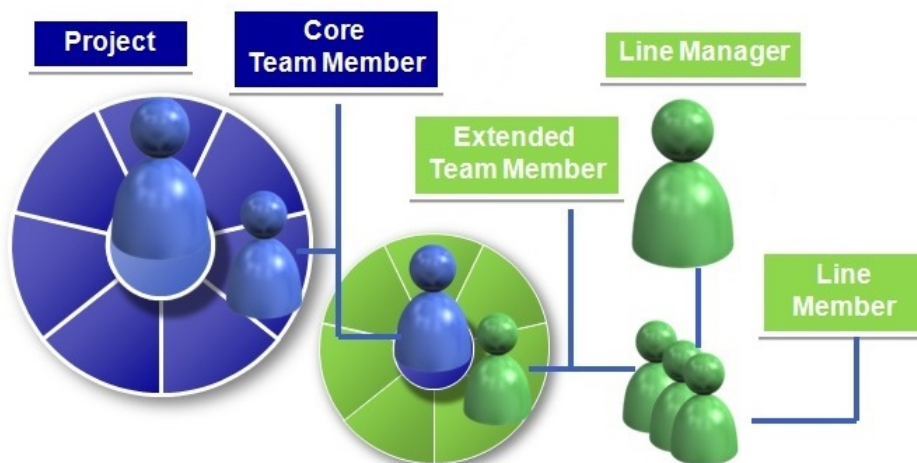


Figure 1.8: Project organization

As can be understood from the above description the V-Model was chosen as process

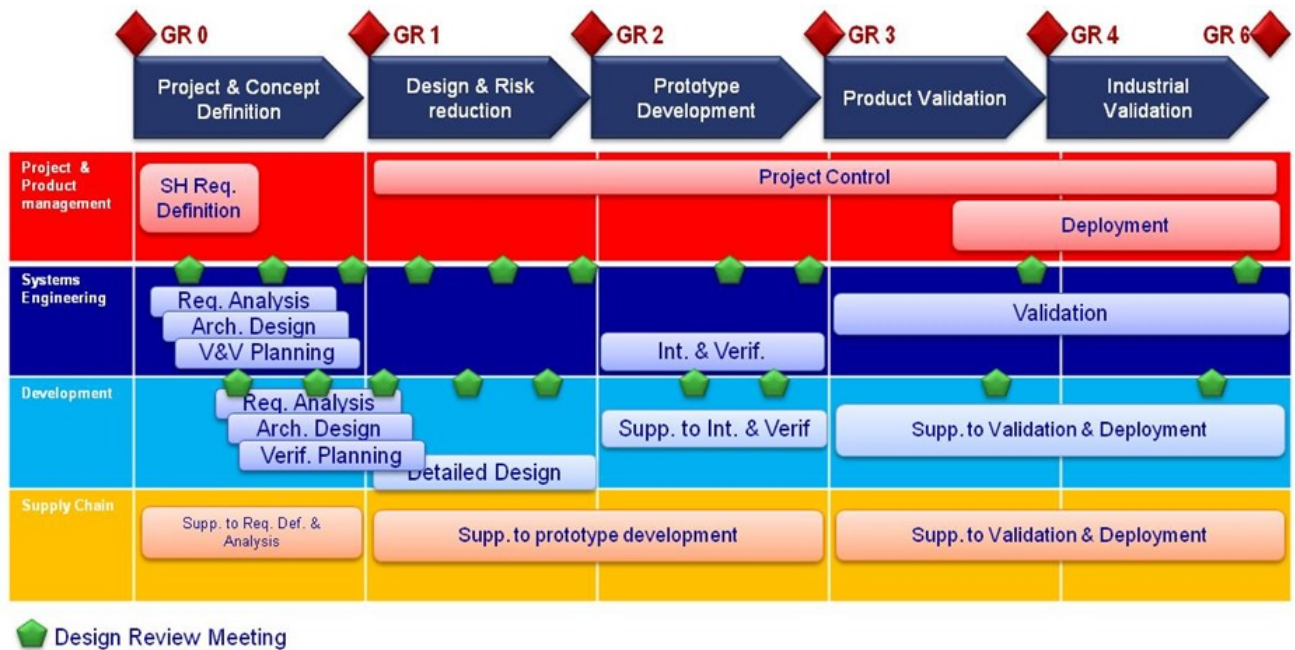


Figure 1.9: Example of PD Process Implementation

framework to support such complex systems development activities. The implementation started some years ago and it is in a continuous improvement phase. The V-Model name comes from the V letter which is used to graphically represent the development process phases through the time (Figure 1.10).

In particular, the left-hand of the V (the descending one) is mainly composed by the requirement analysis and cascade activities, from stakeholder requirements to system, sub-system, module requirements and so on until the desired detail level. Moreover, very important and critical decisions about the system architecture and about the verification and validation criteria are undertaken. The same kind of decisions have to be made at every level, but the root one is the most important because it influences all the decisions at lower levels, it defines a shared vision of the whole system and its features and characteristics. The V peak represents the design activities in their strict meaning, while the right-hand (the V ascending side) is composed by the subsystems verification, system integration, verification and validation activities. The industrial validation and deployment activities are here undertaken as well.

In the graphical metaphor, the descending hand is thicker than the other one. That is to represent that activities have to be undertaken reiteratively in order to minimize error risks. The importance of the decision here taken is so high that, in case of wrong analysis, they could lead to huge money and time losses.

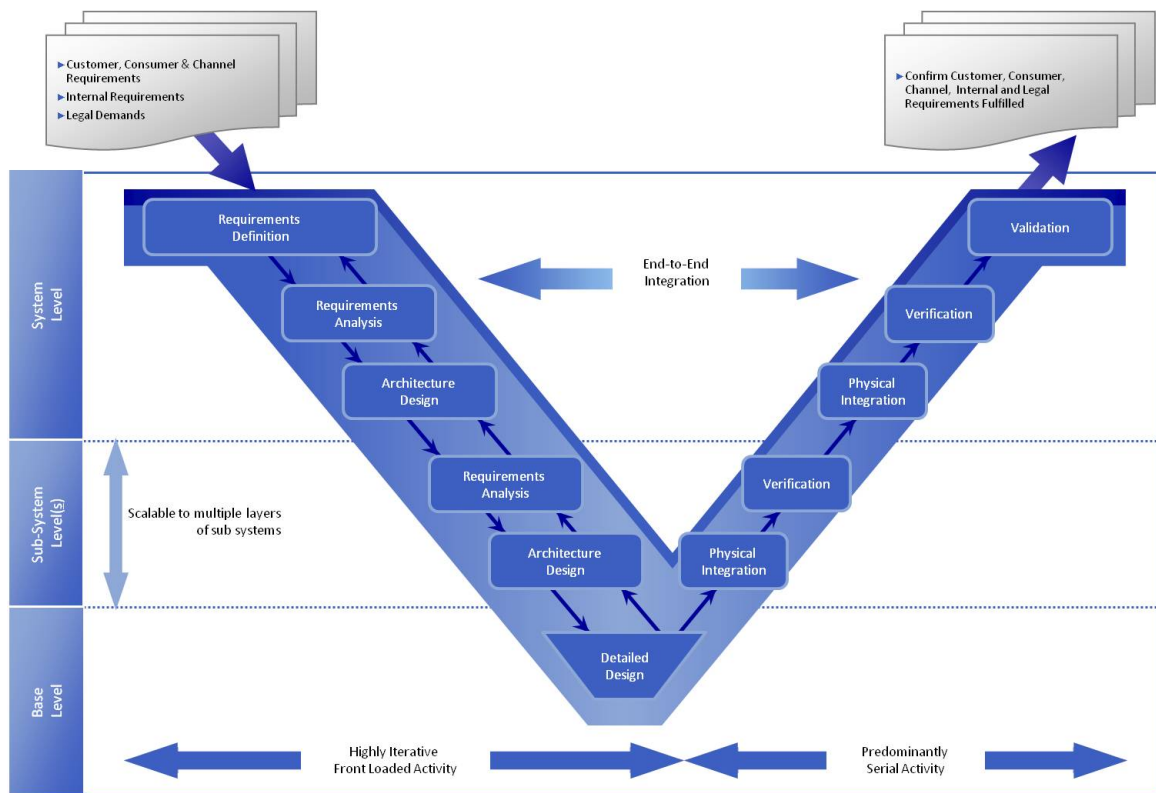


Figure 1.10: Example of V-Model

The list of deliverables was developed in accordance to the V-Model principles (e.g. System Requirements, Architecture Description, V&V Plan, etc.) and some best practices adopted for the specific needs in the food packaging product development activities (food safety assessments, environmental assessments, etc.). Each deliverable was identified as output from a specific activity mentioned in the workflow. Some deliverables were also input to other activities within the same lane or to other lanes, creating an interconnected workflow linking the different lanes.

The implemented process resulted as applicable for any PD project driven in the company. In order to achieve that goal the different projects complexity was taken into consideration by defining some scalability approaches. These approaches allowed the project to scale teams and activities based on its needs and scope.

1.4 Contribution and thesis outline

This thesis focuses on the development of a SE process and related methodologies to support service product business in the packaging company. We started identifying the problem and creating a model of the company by involving strategy, business model, value

chain and products life cycles. According to the existing gate based Product Development process, we decided to use the classic V-Model as framework for the Service Development process workflow. However, according to some stakeholder requirements, we took the opportunity to introduce also some AGILE methodologies in specific phases of the new process. The rationale behind such decision is that we needed to provide a process solution that could be quickly implemented and used by the company's employee without changing the company's governances and organizations. The results of the activity has been applied and implemented during a real pilot service development project aiming at improving customer product quality control process.

In particular, the main contributions of the thesis are:

- A Core Process Map describing what combination of processes the company needs to fulfill its mission. The map is basically showing how the organization strategy is reflected in the company processes by highlighting how the value is created for the customer. The approach was also presented as keynote speech [9] to the INCOSE - Italian Chapter Conference on Systems Engineering (CIISE2014).
- The Service Development process workflow and its deliverables. The existing Product development process has been analyzed, simplified and extended to cover the integration of products and services to create full system solutions [1]. The results are published in proceedings of the 2nd IFAC Conference on Embedded Systems, Computational Intelligence and Telematics in Control (CESCIT), June 2015.
- A methodology that puts in relation the company Value Chain and the product Life Cycle in order to define and manage product requirements extending to service products developments [16]. A paper was submitted and approved for publication in the Journal of Engineering and Architecture (2016).
- A structured approach to manage product architectures to help designers and product architects to correlate the needs of projects stakeholders (requirements) with other needs related to the product strategy and roadmap. The methodology goal is to improve efficiency in terms of resource management, product variants and other aspects that affect the life cycle of the product [14]. This methodology description a related case study can be found in proceedings of the CIISE 2014 - Conference of the INCOSE Italia on System Engineering.
- An AGILE based approach to Product V&V. The practice shows a possible short time verification approach for products that need to be validated directly at the

customer site [12]. A case study was also developed and approved for presentation at the INCOSE - AISE, Verification, Validation and Testing Seminar 2016.

- A validation case study that reports results from the implementation of the process and related methodologies in a service product development project. The project deliverables are both software and capabilities to enable a service to improve customer's products quality control processes [15]. The article has been also published in the Journal of Engineering and Architecture (2015).

The thesis is organized as follows:

Chapter 2 describes in detail the inputs to the activity and the steps that lead from the company strategy and customer expectations to the creation of a Core Process Map update in order to fulfill the expansion and integration of the Service Product Business. The methodology presented in [9] is applied and further detailed in order to update the company core processes by integrating the new Service Development process in the existing Product Development process. Considerations regarding the way to develop measurement systems for the updated process are also presented. The results obtained are integrated into a strong framework and presented in [1]. In the final part of the chapter, will be shown how the presented results were used as input to the activity of process update presented in the following chapters.

Chapter 3 contains the description of the Service Development Process workflow and related main deliverables as described also in [1]. An important method to put in relation the company value chain and the product life cycle was also developed and described in this chapter according to the theory developed in [16]. The presented best practice addresses the problem of identifying stakeholders and related needs when developing a new product or extending an existing one. The chapter reports also some information about a methodology to optimize the management of the product architecture [14].

Chapter 4.2 describes the validation of the developed methodologies in a real case study. The pilot project scope is to develop a service to improve the customer's product quality. The chapter shows all the process steps starting from the requirements to the final validation of the service product [15]. During the project progress description an AGILE approach to the product validation will be described as well [12].

Chapter 2

Core Process Map

2.1 introduction

The development of a process is a critical activity for a company. The output will affect directly resource management, activities organization, deliverables produced and finally the company business itself. Before getting into details on the methodologies and techniques available for the process design, it's important to provide some definitions of "process".

- *"Set of interrelated or interacting activities which transforms inputs into outputs."* (ISO 9000:2005).
- *"A system of operation or series of actions, changes, or functions, that bring about an end or result including the transition criteria for progressing from one state or process step to the next."* (IEEE P1220).

From the first definition it is easy to describe a process like the system shown in Figure 2.1.

- Input: Information and materials to be transformed that are inserted into the process
- rules: instructions, regulations and informations that guide and affect the process activities
- Resources: people, tools, methods and other artifacts to enable process activities
- Output: the result of the process activities

In addition, the second definition explains that another characteristic of a process is the presence of the transition criteria from one process step to another. This characteristic

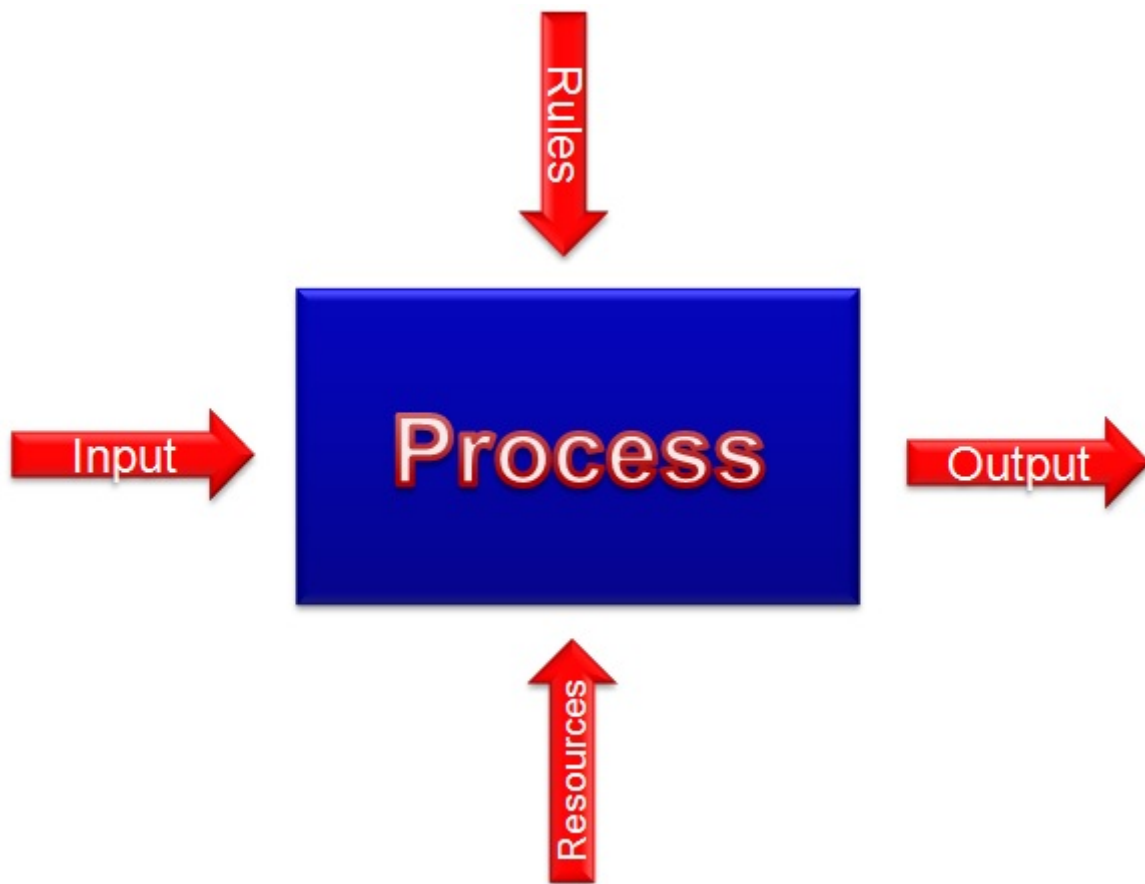


Figure 2.1: Definition of Process according to ISO 9000:2005 standard

is proper of the structured processes [9]. A process is also composed by sub-processes (or sub-level processes) that result to be connected in order to fulfill the process purpose (Figure 2.2)[50].

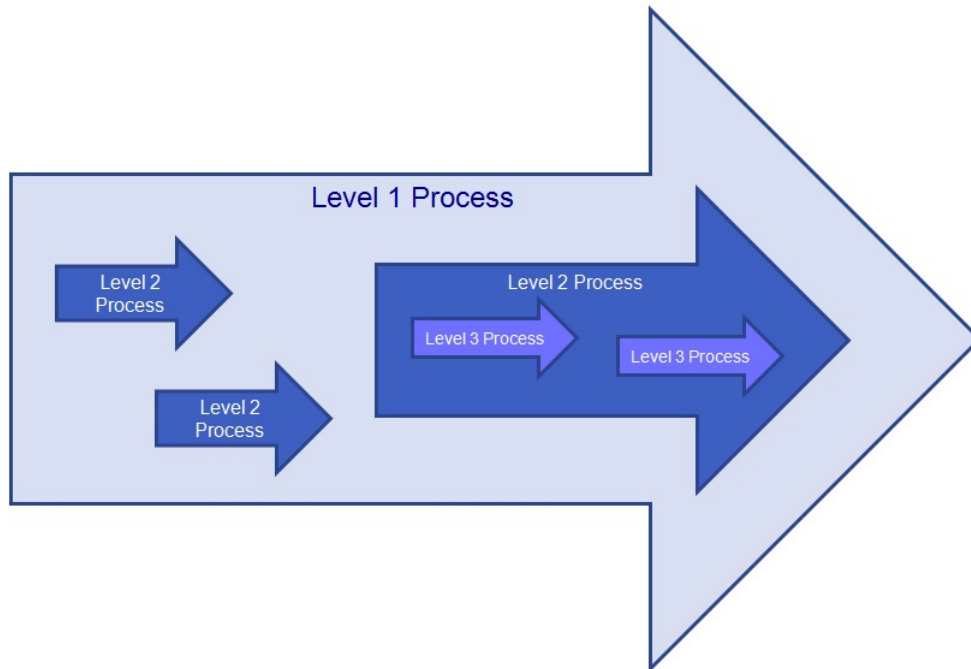


Figure 2.2: A Structured process can be decomposed in lower-level subprocesses

In Figure 2.3 there is a list of typical processes that can be present in a modern company.

Usually the processes can be divided in three main types:

- Strategic Processes: to support all the strategic activities (e.g. Company Strategy definition)
- Supporting Processes: to manage supporting functions (e.g. HR, supply chain, legal, etc.)
- Operative Processes: developed to support operative activities (product development, maintenance & services, sales, etc.)

As written in the beginning of this section the definition of a process is a challenging activity for a company. A lot of methodologies were developed in the past regarding this argument but very few are considering the specific need of reviewing internal processes to expand the actual business by including new services. In this work we adopted the methodology explained in [9] and [27]. The first step in the introduction of a new process in a company is the identification and mapping of the company process under review.

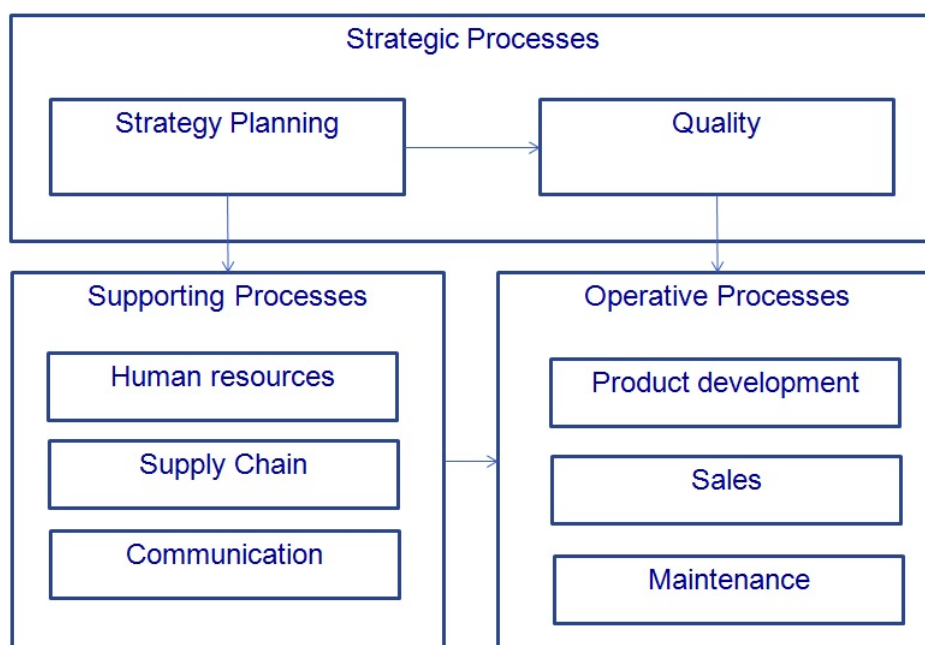


Figure 2.3: Example of typical processes implemented in a generic company

2.1.1 Outline

The outline of this Chapter is as follows. Section 2.2 provides a background on process mapping methodology while the application of the described methodology is shown in Section 2.3.

2.2 Process Mapping Methodology

The definition of a new process or the review of an existing one starts with identification of the beginning and the end of the overall process. In analogy with a system, a good start, to identify the boundaries of the process, is to define its inputs and outputs (Figure 2.1). Another effective approach for identifying the process and how it is constructed is to look at it through the customer's eyes. In this way is easy to understand why a process exists and how much that is critical for the company success[27]. The analysis of the process continues by identifying the process steps (with associated units and tasks) and their connections to lead to the result. Once the process has been defined, it is time to begin talking with people working in the organizations in order to build the Process Map. basically a Process Map is a diagram that contains information about the process that answers the following questions[9]:

- What combination of processes do we need to fulfill our mission?

- How is our strategy reflected in our processes?
- How do we create value for our customers?

A sample Process Map for cheap products producers could be the one showed in Figure 2.4.

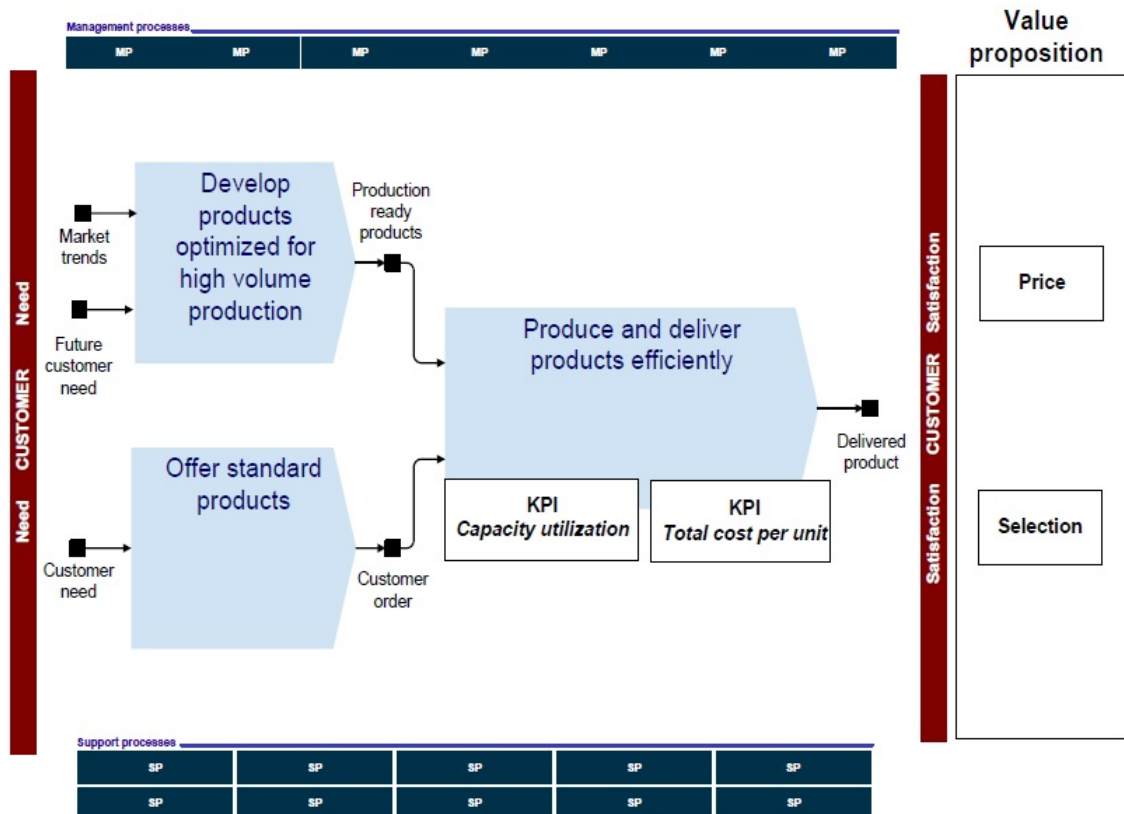


Figure 2.4: Example of Process Map diagram showing the core process for a "Best Total Cost" value proposition.

The examples shows that mapping a core process of a company producing low cost standard products means to identify the following input / output:

- Input: Customer needs in terms of actual needs, future needs and market trends.
- Output: Delivered products fulfilling customer expectations.

From the customer point of view such products shall satisfy certain criteria of selection and price if the company would like to be successful in the market. On the other hand the company, to be profitable, is interested in producing and delivering standard products in

the most efficient way as possible. This aspect is crucial and must be taken into account when designing the process activities and the other supporting processes.

Next section describes the application of the process mapping technique to the company subject of this work and shows the resulting artifact that will be in input to Chapter 3 where the development of the SPD process will be detailed.

2.3 Resulting Process Map

As described in previous section the Process Map technique was used to establish the basis for the review of the Product Development (PD) process and the integration of the Service Product Development process (SPD). The activity started by identifying the reference process for product development. The information was gathered according to the steps described in [27] by interviewing people in the organization owning the PD process:

- Input: Needs (actual and future) and market trends for factories and companies producing packed processed food.
- Output: Leading edge products (equipments for processing and packing food)

In this specific case by interviewing the customer service and the marketing organization we were able to describe the value proposition of the company that reflects the following customer expectation and perspective[9]:

- Time: Products must be delivered on time and before competitors.
- Functionality: A leading edge product must contain new functionalities and technologies.
- Brand: With the product the company delivers brand trustability.

The resulting PD process map is shown in Figure 2.5.

It is clear than the analysis activity was conducted by following two dimensions:

- Top-down by looking to the company strategy in order to get the strategic perspective.
- Outside-in by looking to the customers expectations in order to get their perspective.

Interviewing the company's Top-Management, the company strategy involving the service products development was clarified by the following key points:

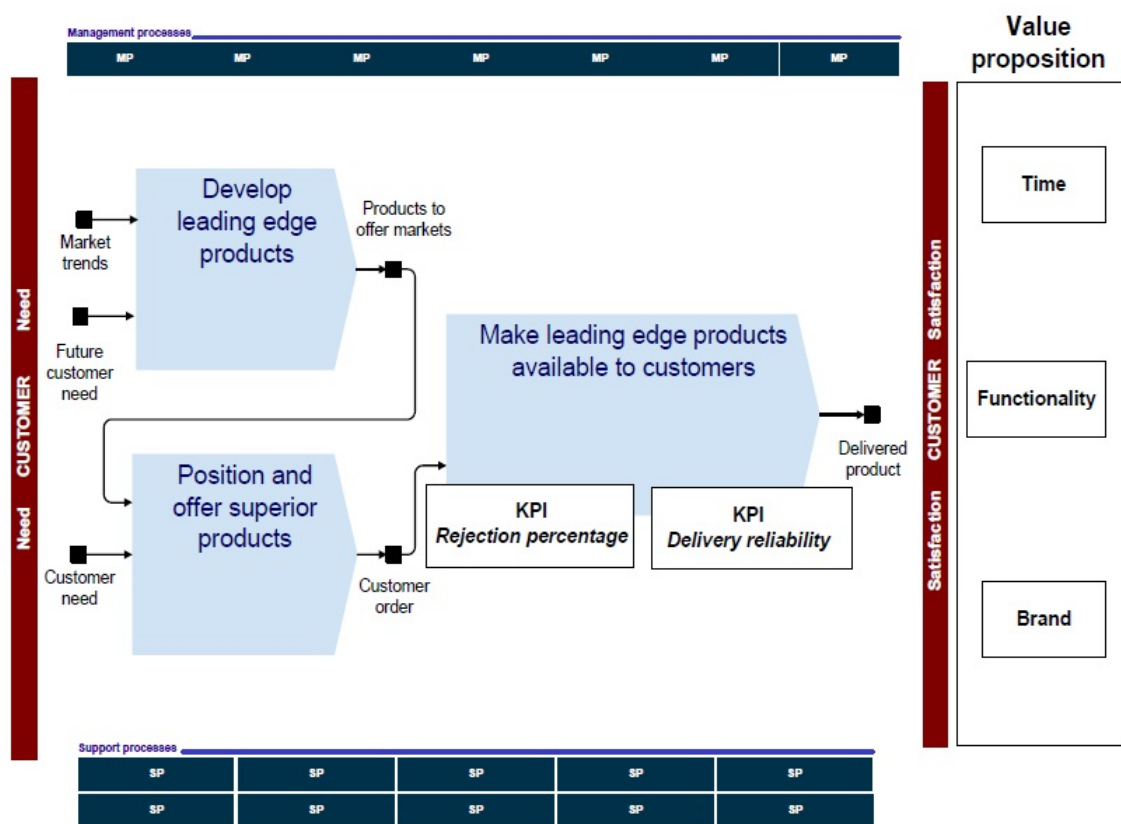


Figure 2.5: Resulting Process Map for the Food Processing & Packaging Product Leader

- Expand business by associating services to equipments.
- Keep in mind the company core values of customer focus, quality & innovation, responsibility and food safety.

Since each organization has its own logic for how to break down strategy into processes and products we decided to perform interviews to representatives of possible process and service product stakeholders. After the interviews we were able to confirm the types of service products that the SPD process shall produce [46].

Service Products that are the output of the process, are characterized by focusing on the support of customer activities related to the use of the product (Figure 2.6).

The first main category is **product-oriented** services. Here, the business model is still mainly geared towards sales of products, but some extra services are added. The second main category is **use-oriented** services. Here, the traditional product still plays a central role, but the business model is not geared towards selling products. The product stays in ownership with the provider, and is made available in a different form, and sometimes shared by a number of users. The last main category is **result-oriented** services. Here, the client and provider in principle agree on a result, and there is no predetermined product involved.

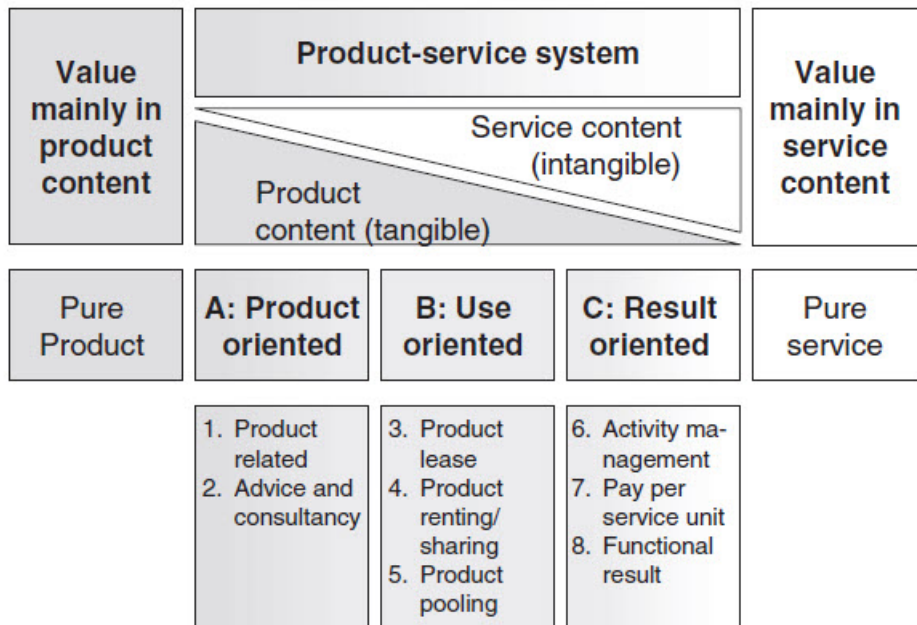


Figure 2.6: Type of Service products to be produced by using the SPD

There are several different definitions of service found in the research literature. Among the most useful are[36]:

- To place a bundle of capabilities and competences (human, technological, organizational) at the disposal of a client and to organize a solution which may be co-produced with the client and given to varying degrees of precision, codification, and customization[13]
- Service products are something a customer pays for receiving even though it may be intangible. The service may be attached to a tangible product however, although a great many service products are intangible even though they may have a physical manifestation. Often, where customers and employees are in relatively constant contact, inter-personal experiences are critical to the delivery of service products[45].
- A service is an activity or series of activities of a more or less tangible nature that normally, but not necessarily, takes place in interaction between a customer and service employees and/or physical resources or goods and/or systems of the service provider, which are provided as solutions to customer problems. Advances in information technology increasingly may remove the person-to-person interaction as a service is provided. Examples of this are the ubiquitous deployment of ATMs and more recently, check-in terminals at airports and hotels[17].

The new developed process map (Figure 2.7) is conceptually and operatively completely different than the one shown in Figure 2.5. The new process map still gets as input the needs of the customer that, this time are related to services that address the entire life cycle of the product to which they are associated. The need is then elaborated by sub-processes in order to understand it and develop the service solution that is delivered to the customer. The new process map also indicates the new value proposition that from the customer point of view results[9]:

- Service: The new product is not anymore only an equipment. The customer buys an integrated solution where the equipment is enriched by a series of additional products that improve its usage.
- Relationship: By hearing to customer needs and problems related to the use of the product, the company increases fidelity and improve the capability to satisfy the needs of the new customers.
- Brand: Trustability becomes even more important in this case since the company delivers not only the capability to build physical products but also its expertise and experience in the use of the product.

2.3. Resulting Process Map

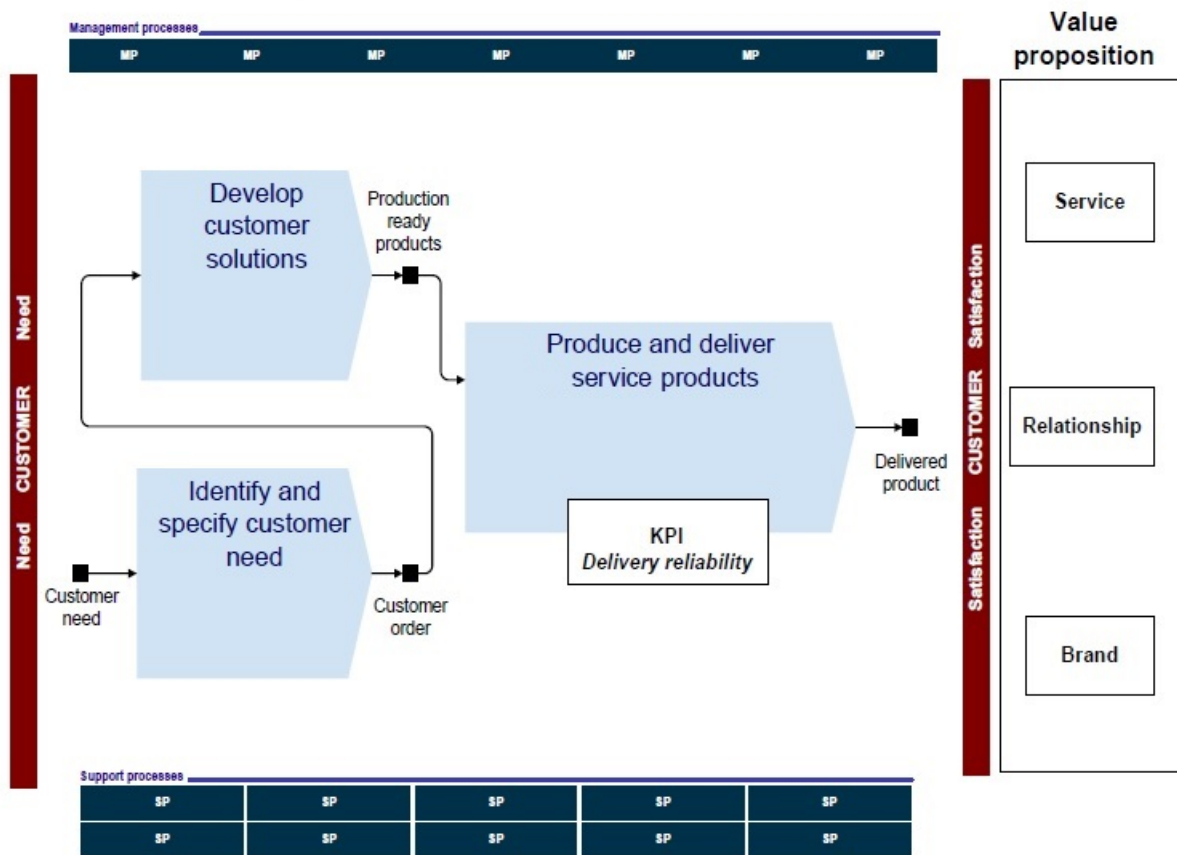


Figure 2.7: Resulting Process Map for the Service Product Process

2.4 Discussion

The Mapping of a process is a critical activity both for process updates and new process design. If this activity is not performed in a correct and structured way there is the risk of process inconsistencies and unclarity in the value creation flow. The process map helps bringing clarity and structure to companies that would like to define their own processes and agree on each level of its internal organizations on the value creation process. In this chapter two different problems are addressed by the Process Mapping technique. The first problem is the assessment of the existing PD process. The process map is really effective to clarify the flow of the information, input, output of the process and the customer's expectations (value proposition) to the product. The second case is the design of the SPD process map. In order to get all the information, some interviews were performed to the different product and process stakeholders. The strategy of the company is used as input and a new value proposition is described. The service product is identified as belonging to one of the three main archetypes of services. A resulting process map is then developed and available for breakdown into subprocesses.

Chapter 3

The Service Development Process

3.1 introduction

The backbone activity of every industrial process that aims to produce an artifact, including mechatronic devices and packaging machines, is formed by the Systems Development LifeCycle (SDLC). SDLC is the process of managing the entire life cycle of a product development from its conception, through design and manufacture, to deployment. It begins from the collection of the stakeholder needs and their translation on system requirements and terminates with the release of a real system. Several SDLC methods for supporting the design of mechatronic systems have been theoretically developed and some of them were subsequently introduced in real applications.

Boggs, in [4], presents an introduction on the SDLC methodologies, including the *incremental model*, the *spiral model*, the *Win-win spiral model*, the *V-model* and the *W-model*, concluding by some thoughts on the application of Six-Sigma concept to SDLC (Figure 3.1).

All these framework have two main characteristics in common:

- Structured sequence of activities: each framework present a certain number of activities that are connected each other by deliverables. The activity order can't be changed otherwise the risk of failure of the final product could be high.
- Each activity can be classified according to the standard types (Analysis, Development, Verification, Validation and Deployment).

Market pressures demand that the systems leverage on technological advances to provide a continuously increasing capability at reduced costs and within shorter delivery cycles. The increased capability drives requirements for increased functionality, interoperability, performance, reliability, and smaller size and determined the born of the

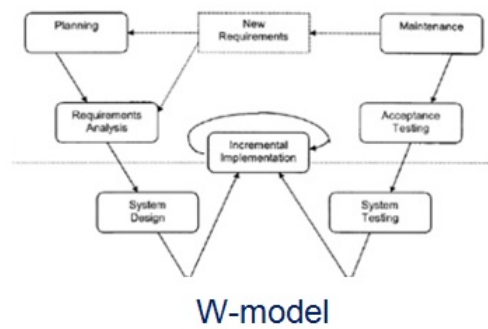
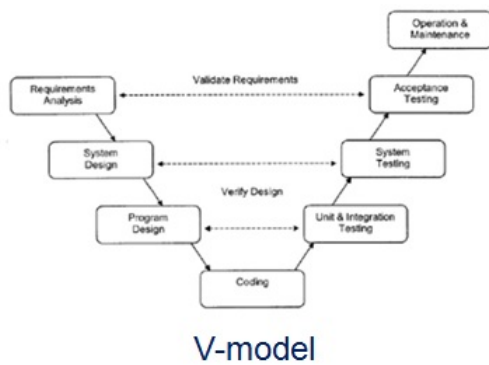
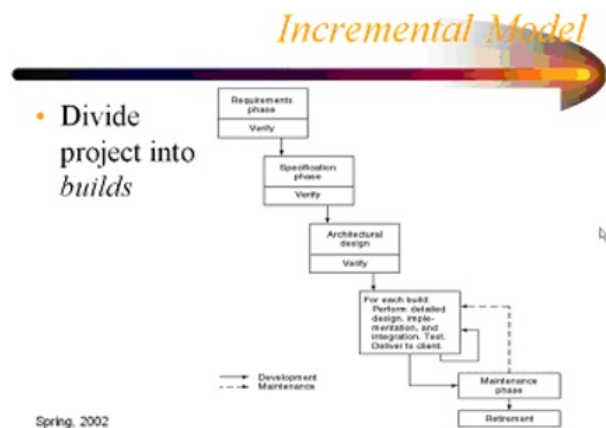
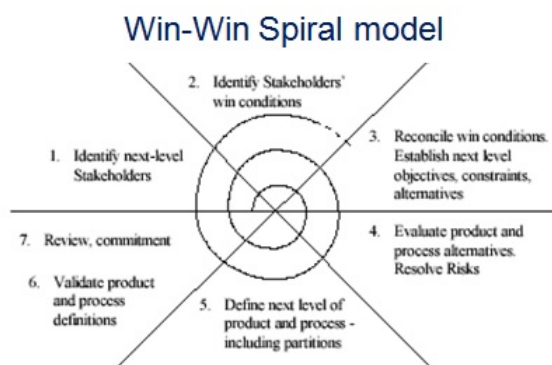


Figure 3.1: Example of different process framework

mechatronic systems. The design and development of these systems requires a multi-disciplinary approach through the integration of mechanical, electrical, electronics and software engineering. Each discipline focuses on a particular aspect of the system and exploits different domain-specific models that should be merged and integrated. In order to cope with this situation several tools implementing different capabilities are commercially available and start to have quite a large diffusion among industrial development communities. However, the implementation of tools is not enough to develop performant mechatronic systems, but optimal industrial processes are necessary. In fact, although the importance of innovation is fully realized by most enterprises and they continue to spend more and more on innovation, many of these initiatives do not generate satisfactory profit or competitive advantage, [5]. The problem does not lie in the invention part or the generation of innovative ideas, but more in the successful management of the innovation process from an idea to a successful product in the market: there is a lack of effective design processes, [28] and [30]. The implementation of structured frameworks in mechatronic products design lets designers to get an holistic view of the problem to solve. In addition the process' top/down approach allows to take into consideration the "big picture", the customers needs and the solution that will be delivered reducing risks and claims.

As already mentioned in the introduction the case described involves a Multinational Company developing full systems and solutions for the food packaging market. The V-Model was chosen as the most suitable development process because of the project based development setup, the size and the organizational complexity of the company. The company setup requested a strong and present governance that led the most critical project decisions with a step by step control of the progress and with a structured gate based approach. After a trade off analysis, the V-Model was also chosen as the most suitable framework for the new process because of its capability to keep activities under control without requesting a high level of experience from users. The implemented PD process (Figure 1.9) led to to a reduced Time To Market (TTM), a more efficient process scalability approach and an improvement of the products quality[1]. However as already mentioned in Section 1, the company decided to expand its business by including a series of service products in its product portfolio.

This chapter will focus on:

- The detailed description of the Service Products Development (SPD) process including its governance body.
- A best practice that puts in relation the product life cycle with the company value chain to support the requirements definition activities.

- A methodology to manage product architecture during the solution definition phase.

In **Chapter 4.2** the validation of the developed methodologies in a real case study are shown. The pilot project scope is to develop a service to improve the customer’s product quality. The service concept has been validated with the company customers and some results will be mentioned as well.

3.1.1 Outline

The outline of this Chapter is as follows. Section 3.2 provides the description of the SPD process as result of the process design activity. The Value Chain vs product life cycle best practice is shown in Section 3.3 while the product architecture management is described in Section 3.4.

3.2 The Service Development Process

The developed SPD process map that was described in previous chapter (Figure 2.7) was used as input for the SPD design activity (Figure 3.2).

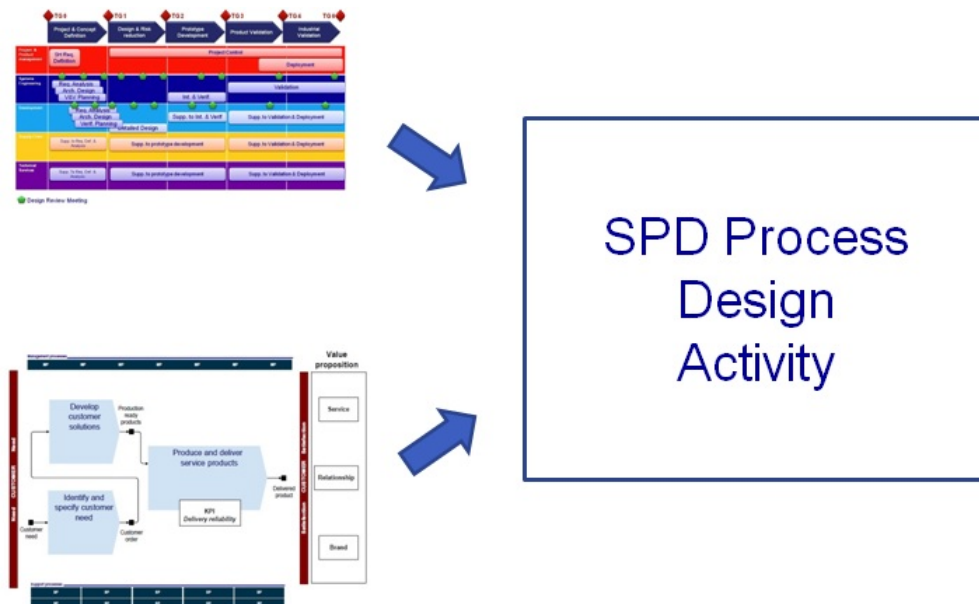


Figure 3.2: The PD process workflow and the SPD process map as input to the SPD design activity

In order to quickly implement and make operative the new process the company supported also the idea to keep the same framework and model setup of the PD process

(same concept of gates, governance, phases and activities) [1],[44]. The design activity then consisted in a breakdown of the SPD process map in four main process phases (Figure 3.3):



Figure 3.3: The four main phases of the SPD process

- Discover: where the service project scope is clarified.
- Define: in this phase the service solution is defined and ready to be implemented.
- Develop: where the service is developed.
- Deploy: after this phase the new service product is deployed to the market.

In next subsections a detailed analysis of each single phase will be showed but, before entering into details of what need to be done in terms of process activities, it is better to show who is involved in a service development project. According to the company setup and way of working, a project manager is appointed as project responsible before the start of the project. Similarly to the PD process the project manager is responsible for the entire project execution and related deliverables. In general the project manager's tasks are [23]:

- Project scope definition
- Resource allocation and management
- Budget management
- Project planning

In addition to the above tasks, the project manager is also responsible to bring the project recommendation on project progress to the project governance. The project governance is composed by representatives of all the operative organizations of the company and in general the following ones are represented by permanent members :

- Production

- Marketing
- Supply Chain
- Customer Service

In addition to the first three whose function definition has been already provided in Chapter1, The Customer Service organization was included. This company function is responsible to manage customers by supporting them with the company services and reporting back to the organizations the customer's feedback. The project governance has the task to assess the project progress in terms of budget, time and quality. The governance has also the power to stop a project if that does not fulfill one or more KPIs (Key Point Indicators) related to the above three criteria[9]. A service product development project needs to involve also a marketing representative to work in the project team. the motivation is easy to be understood since such products are critically dependent on the customers needs and market trends. Following list shows typical tasks of a marketing project member[23]:

- Customers and marketing needs definition
- Business case definition
- Manage deployment plan
- Manage communication plan

The service solution is developed by the technical project team member. Usually this person lead a team of resources whose efforts deliver the service to the project. In general the identified tasks are[1]:

- Solution analysis
- Solution development
- Solution verification and validation

3.2.1 Discover phase

The Discover phase is the first phase of the SPD process. It starts with GR-I (first gate review) and in analogy with the V-Model is identified in the analysis phase (Figure 3.4). During this phase:

- **Project Management:** Defines the scope of the project, allocates budget and makes a project plan by identifying resources activities and tasks. In addition the project manager collects requirements from internal stakeholders (e.g. legal, quality, environment, etc.)
- **Marketing:** Defines the business case for the service product and provide requirements from customers and market to the project team.
- **Solution:** Analyzes and develop the solution requirements in order to prepare for the solution definition. The requirements analysis activity is a critical activity in each type of projects and especially in service development. In order to reduce the risk of not fulfilling the customer and stakeholder needs the Value Chain vs Life Cycle Approach for Product Extensions has been developed and will be detailed in Section 3.3. An initial concept screening is also performed in this phase in order to select some alternative concepts to be translated in service solutions.

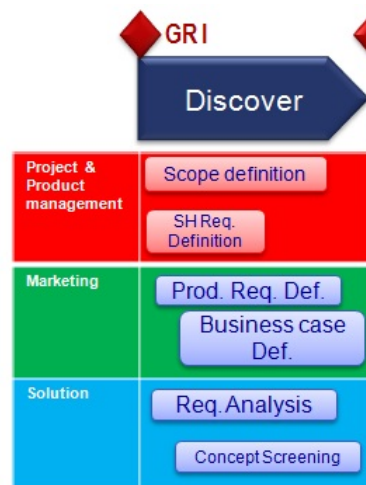


Figure 3.4: The Discover phase of the SPD process

From a technical point of view the output of this phase is a collection of requirements at system level and a series of reference concepts.

3.2.2 Define phase

If the service product project successfully passes the GR-II, the project enters in the solution definition phase (Figure 3.5).

During this phase:

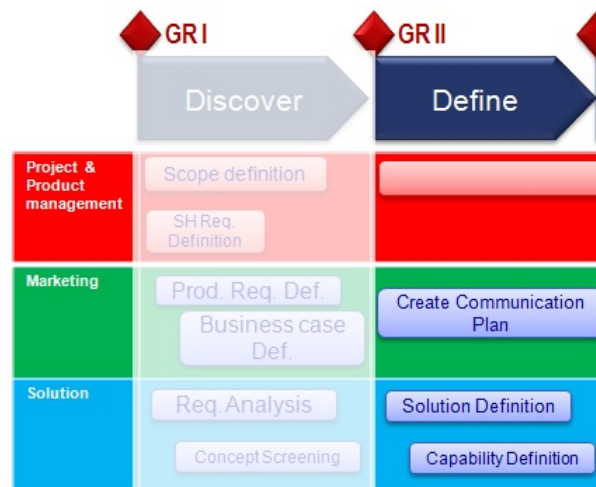


Figure 3.5: The Define phase of the SPD process

- **Project Management:** Keeps track of project progress, updates plans and budget and supports the other project members in their activities.
- **Marketing:** Creates the communication plan for the new product launch. Identifies key challenges and opportunities related to product launch communication and propose the communication strategy.
- **Solution:** Define the product functional, physical and operational architectures based on the defined requirements. Define and describe in engineering terms the interfaces between sub-systems. Define required capability elements for running the service in terms of specific competencies of people, tools, processes or procedures needed.

After this phase the complete service solution is defined on the paper and is ready to be transferred to next phase when the solution will be developed.

3.2.3 Develop phase

After the GR-III, the project enters in the solution development phase (Figure 3.6).

During this phase:

- **Project Management:** Actively monitors the project progress and collects input for requirement changes and updates, from all stakeholders.
- **Marketing:** Executes the product launch communication plan to validates how it work. Align activities with pilot markets and customers. Ensures stakeholders understand product. Finalizes product communication material

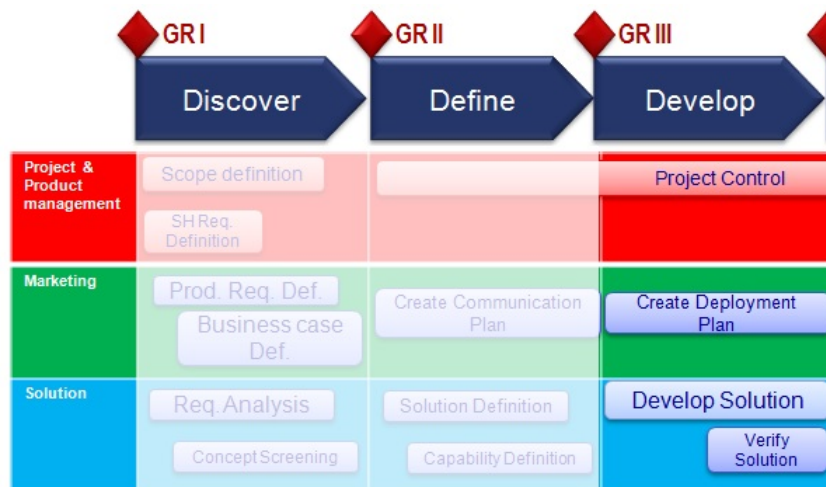


Figure 3.6: The Develop phase of the SPD process

- **Solution:** When a project has gone through the Develop phase the building blocks of the service are developed. In many cases, the building blocks composing a service are a combination of existing and new blocks, and in this phase the new blocks are developed and the existing are secured that they fit into the new context of the service (integration). The solution is verified against detailed requirements (possibly through external verification).

At GR-IV, the documented deliverables are detailed and the expected maturity in the documentation is high. All documentation and solution regarding the service should be available in a stable version.

3.2.4 Deploy phase

After the GR-IV, the project enters in the solution deployment phase (Figure 3.7).

During this phase:

- **Project Management:** Secures that receivers of the product take on their responsibility and also start to be part of the deployment and play an active role during the ramp up. Secures also that the new product will be managed delivered and sold according to the defined requirements.
- **Marketing:** Executes according to the finalized deployment plan for global release of the product.
- **Solution:** Generally this is done through running several deployment projects with different characteristics and at different customer archetypes or with different build-

3.2. The Service Development Process

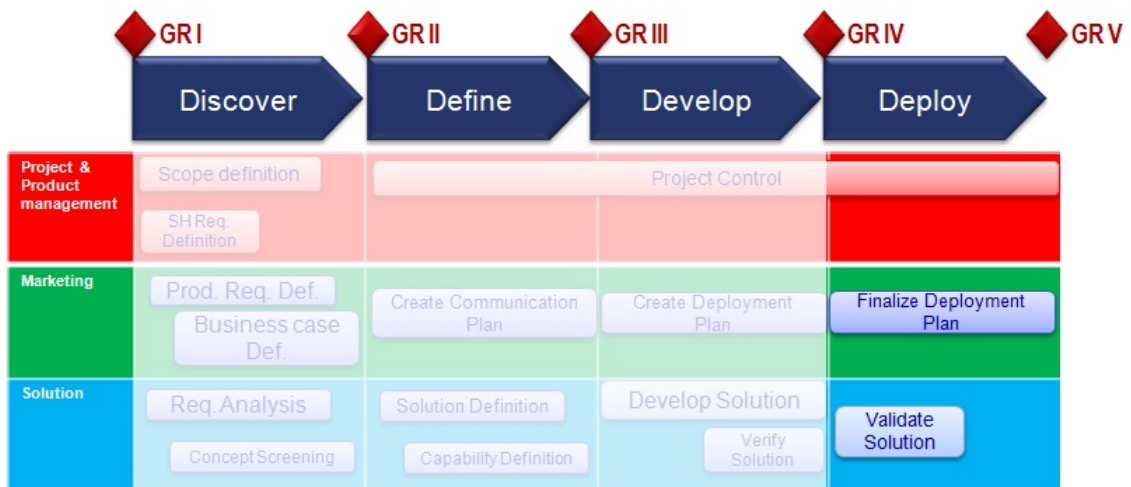


Figure 3.7: The Deploy phase of the SPD process

ing blocks of the service (different scope). These projects validate the service from the full perspective of Sell, Deliver and Manage.

After the GR-V the Service is inserted in the product portfolio and the technical solutions are transferred to their final owners (e.g. Customer Service organization, Sales, etc.). The full picture of the SPD process is then shown in Figure 3.8.

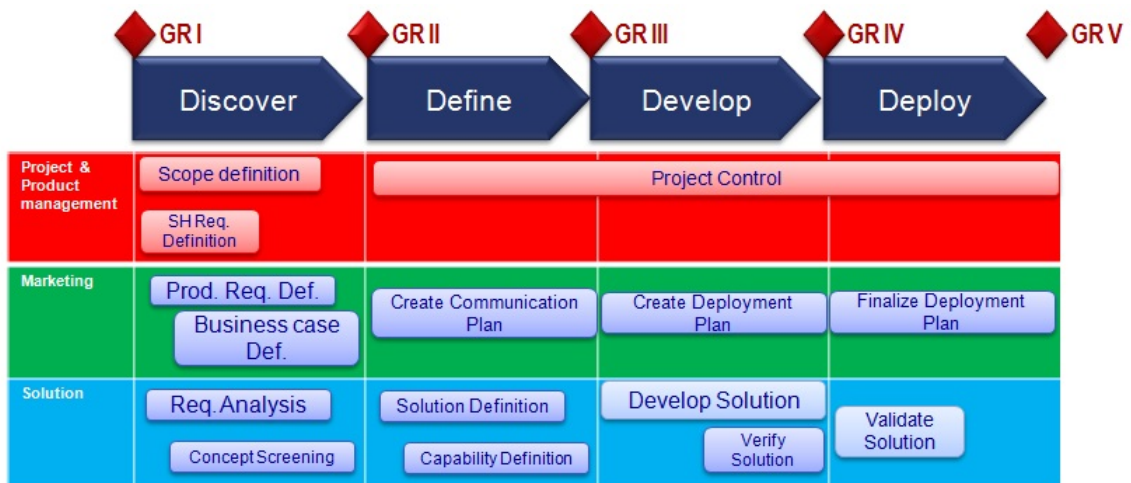


Figure 3.8: The SPD process workflow

3.3 The Value Chain vs Product Life Cycle Approach

As mentioned in this chapter introduction a methodology to support requirements definition has been developed to improve the capability to extend a physical product (e.g. equipment) by creating a full system/service solution. We know that an artifact that is produced by a company and fulfills needs of some stakeholders is called Product. It consists of a series of interconnected features or functionalities and is identified within a context and applicable in specific value chain. A product, like any other evolving thing, is characterized by a life cycle that starts with the conceptualization and development and finishes with the decline phase passing through growth and maturity (Figure 3.9). As shown in the figure the maturity phase is usually characterized by a maximum in both profit and sold units. Thus companies are very interested in keeping this phase as long as possible. For this reason in the maturity phase it is possible to see a product renewal that can consist of extensions of the product functionalities/features.

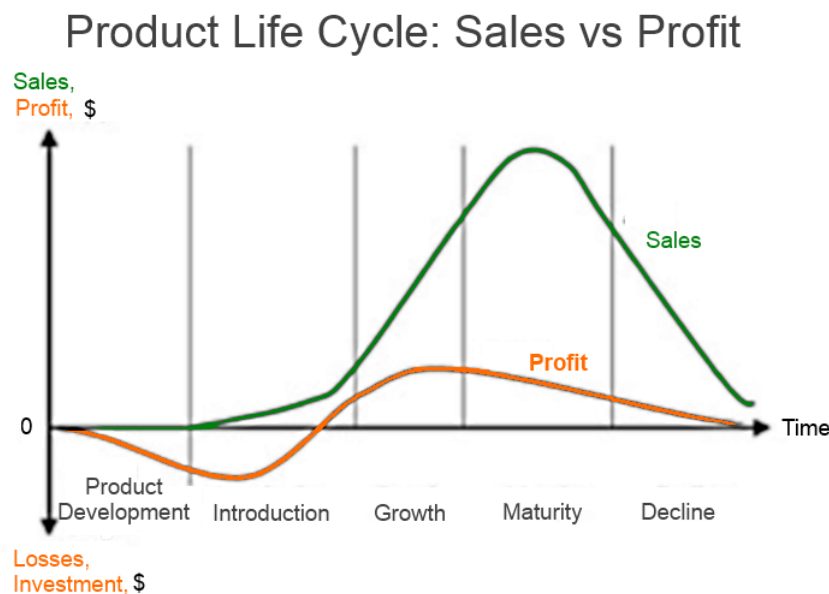


Figure 3.9: Example of Product Life Cycle from profit and sold units perspectives

The product extension is usually needed because even if the general product concept could remain valid from a marketing perspective, the change comes from the market in terms of new requirements or needs from customers (e.g. improved performances, side functionalities, user or technology maturity, services, etc.). The market requirements are the starting base to make considerations on the product evolution. However a structured engineering methodology is useful to secure that all the requirements from all the identified sources are captured and addressed.

The presented methodology aims at enabling the analysis and development of a list of criteria and drivers to understand the future evolution of the product. The methodology is requirements driven and puts in correlation the product life cycle phases and the company value chain. The output of the methodology application is a list of requirements mapped on the company value chain and product life cycle. From the analysis of the map it is possible to derive the product evolution and to address the solution to the right direction. The extended product shares with the parent one the same reference architecture and functionalities. Thus this methodology might be integrated with others for the Architecture Management of the product [1]. In the following subsections the methodology is described while for the application in a real industrial case, the interested reader can refer to [16].

3.3.1 Methodology Description

The introduction of a new product or the extension of an existing one is usually driven, from a generic point of view, by the marketing department. They provide the list of needs that usually in later stages of the product development activity are translated into requirements. The main source of those inputs comes from customer, market and competitor intelligence activities. In order to have the full list of needs, other points of view should be taken into consideration (e.g. Customers service, legal department, supply chain, etc.).

The proposed methodology helps in providing structure to the needs mapping into the Product life cycle phases while considering different points of view in the company value chain. It is composed by the following macro steps:

- High Level Needs identification
- Extendable Platform Product choice and its relevant life cycle stages identification
- Stakeholders matching with LC phases
- Needs further detailing in requirements

High Level Needs Identification

The first stage of the methodology consists on the identification of the needs (internal or external) that are the reason to start the product extension. In general the needs considered in this stage are results of marketing department activities aiming at scouting and analyzing new business opportunities or strengthening the existing ones. The result

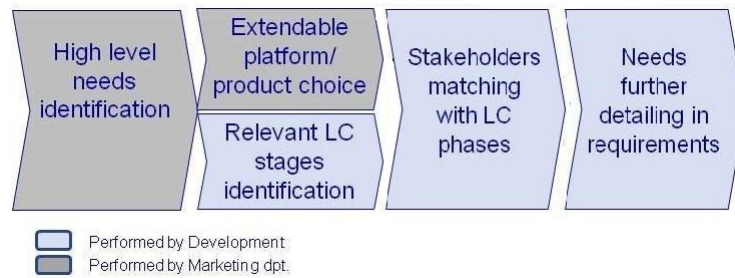


Figure 3.10: Workflow description

is created by using marketing tools like interviews, scouting of competitors or similar businesses (e.g food and pharmaceutical for packaging business), business intelligence simulations for trends, etc.

Extendable Platform Product Choice and its relevant life cycle stages identification

In this stage the methodology is divided in two sub-activities:

- Extendable Platform Product Choice
- Relevant life cycle stages identification

The first activity is simple in theory but critical from the company point of view. In fact in this activity the project selects the platform of the product to extend. The choice is really dependent on how the company organized its products and related portfolio as well as its development operations. The output of this activity is a list of stakeholders of the platform (Figure 3.11). It is important to highlight that by selecting the platform and generating the list of stakeholder we are implicitly considering the company value chain.

The second activity is the identification of the relevant life cycle stages of the product. We are speaking about “relevant” because maybe not all the product life cycle stages could be involved in the product extension. In Figure 3.12 the common product life cycle stages are indicated.

Stakeholders matching with LC phases

In this step we put in relation the product life cycle stages with the stakeholders in order to understand and match where in the life cycle of the product there is a stakeholder’s need to be documented. The result is the SHLC matrix in Figure 3.13.

3.3. The Value Chain vs Product Life Cycle Approach

- Customer
- Consumer
- Channel
- Environmental
- Legal & IP
- Package Safety
- Equipment Safety
- Food Safety
- Hygienic Design
- Export Control
- Equipment Supply Chain
- Add. Mat Supply Chain
- Pack. Mat. Supply Chain
- Base Mat. Supply Chain
- Package Development
- Base Mat. Development
- Add. Mat. Development
- Pack. Mat. Development
- Equipment Development
- Technical Service
- Market Company
- Strategy & Planning

Figure 3.11: Example of product platform stakeholders

- Common Stages**
- Propose
 - Develop
 - Procure
 - Sell
 - Supply
 - Install
 - Operate
 - Maintain
 - Dispose

Figure 3.12: Example of common product life cycle stages

	Consumer	Environmental	Export Control	Base Mat. Supply Chain	Technical Service	Market Company
1 Propose				x		x
2 Develop	x	x		x	x	x
3 Procure				x		
4 Sell	x		x			x
5 Supply	x		x			
6 Install						
7 Operate	x				x	x
8 Maintain		x			x	
9 Dispose		x				

Figure 3.13: Example of SHLC matrix

Needs further detailing in requirements

Last part of the methodology supports the need collection in order to further translate them into requirements. A sample need list can be seen in Figure 3.14.

Consumer		
Develop	Need 1	The product needs to be pretty.
	Need 2	The product needs to be lightweight.
	Need 3	The product needs to have a long shelflife.
	Need 4	The product needs to be robust.
Sell	Need 1	The product needs to be reasonably priced.
	Need 2	
	Need 3	
Supply	Need 1	The product needs to be easy to stack
	Need 2	
	Need 3	
Environmental		
Develop	Need 1	The product needs to be recyclable.
	Need 2	The product needs to be non-toxic.
	Need 3	

Figure 3.14: Example of needs captured from the customer correlated with the LC stage of the product

Once the list is complete the project can start analyzing each need and develop the proper requirements linked to it. This simple best practice is really effective for new products because it gives a guide that avoids to loose information during the needs collection before developing the project requirements. The approach is also very useful for product extensions or to develop new services to be integrated in existing products, because also in this cases a guiding tool to capture information and structure the requirements development activity in provided and can contribute to reduce the risk of inconsistent or non relevant information.

3.4 Product Architecture Management - an approach to Product Life Cycle

Complex products are integrating very different functionalities that are belonging to different technologies and disciplines and must coexist all together in order to achieve the common goal that is a successful product that brings value to the customer and to the company. In order to achieve the above goal a structured company follows different processes to acquire the input needed from the customer, translate them into requirements that are satisfied by the implemented solution. This implies an organization structure

that usually consists of projects (responsible for the product delivery) and a set of line organizations (responsible for the product development).

The methodology developed will help product (system/subsystem) owner or responsible to efficiently develop a solution roadmap that will optimize the resource allocation and the releases to the related projects.

As mentioned in the in Section 1.3, usually, a common operational model consists of a project entity (a team lead by a project manager) and one or more development lines (teams lead by line managers). The project is responsible for the acquisition of the requirements, control of the planning and deliveries and on the other hand, the lines are responsible for the design of the solutions that will fulfill the requirements received by the project (Figure 1.8).

The implications of such organization is that the increasing number of projects is not always increasing consequentially the number of resources available from the lines that have to develop the solutions. So a certain resource have to work in parallel in more than a project. In addition a line have to manage different sets of requirements belonging to different projects that sometimes are conflicting each other or causing reworking due to sequential update of certain functionalities in a product. The detected resulting situation causes big problems in managing the versioning of a certain architecture of product (or component) increasing thus the effort in developments and configurations taken by the lines that are owning it.

To support lines in the management of a product a lot of methodologies are available in literature [49], [48], [19]. Most of them consists of prioritization and classification of requirements performed by the company's marketing organizations or development governances. The outcomes support the decision of what features to include in a certain product. The methodology described in this chapter improves the product architecture management by enabling the line to actively participate in the decisions regarding the solution evolution by directly owning its roadmap.

3.4.1 Background

An ideal process of product development and management starts with needs of customers. They are analyzed by the company market organization which then creates a set of stakeholder requirements. That list together with other internal stakeholder requirements will be satisfied by a project with the delivery of a solution. As already mentioned large companies, usually, are involved with several concurrent projects and produce different configurations of their products. The worst possible case of the above situation is when a development line delivers one version of its product per project. Assuming that each

version may be considered as a stand-alone product, in case of optimized architecture management, every version is the result of the new features addition to the older ones (case n.1). In case of unoptimized architecture management there is the coexistence of more than one concurrent version in the same life cycle phase of the product (case n.2). A simple example (Figure 3.15) that explains the above situation is related to the product user manuals describing the features added to the different product versions. In case n.1 the line has to manage one user manual evolving together with the product versions along with the product life because the latest version is able to replace (includes) the older ones. Three different user manuals have to be generated and separately managed in case n.2 due to the partial dependencies between the different product versions.

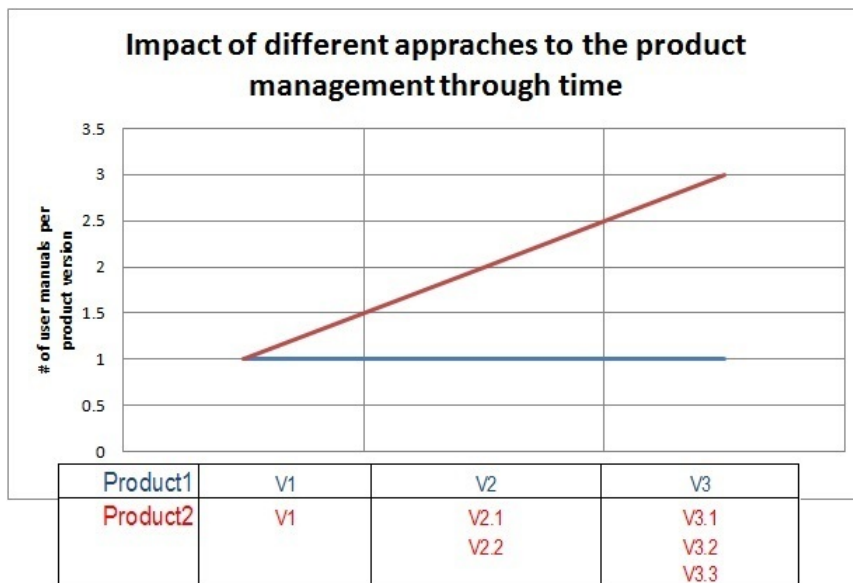


Figure 3.15: Example of number of user manuals per product versions in case of two different product management approaches

The intention of the example is to provide an indication of how much complex could be managing different "stand-alone" concurrent versions of the same product. The effort for such management increases proportionally with the number of existing concurrent versions causing the need for the line to ask for more budget and resources to keep maintaining those solutions.

3.4.2 Methodology Description

The methodology can be applied used in any type of development project and is fully applicable for any type of product (e.g. service products, mechanical equipments, mecha-tronic systems, software, etc.).

In the starting situation a list of projects (and related requirements) that the product has to fulfill must be available. The product should provide also some supporting documentation that describe its design and related technical decisions.

The applied methodology steps are described in the following list:

- Requirements vs Projects review
- Architectures definition
- Release Versions Definition

The first step rearranges requirements together with project in order to provide the full picture of what the product has to fulfill. In the Architecture definition task, product commonalities and variants are defined according to projects and products families where the system under development could be applied. Last step defines the product roadmap by identifying its release versions (Figure 3.16).

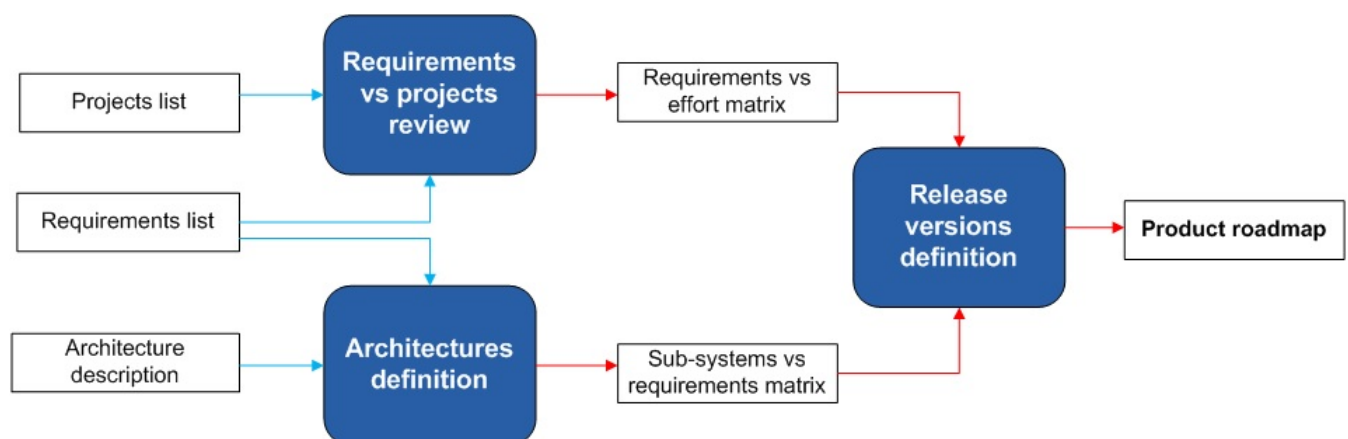


Figure 3.16: Methodology workflow with input and output

Inputs

To start with the activity the complete list of project (and related time-plans) is required. The list consists of all the projects (both on going and future) that are involving the product providing in this way a three years view of the system developments (Figure 3.17).

Together with the project roadmap the full list of requirements per each project is requested as well. All the requirements could be independent each other, identifying thus one feature each. No priority could be defined yet to those requirements.

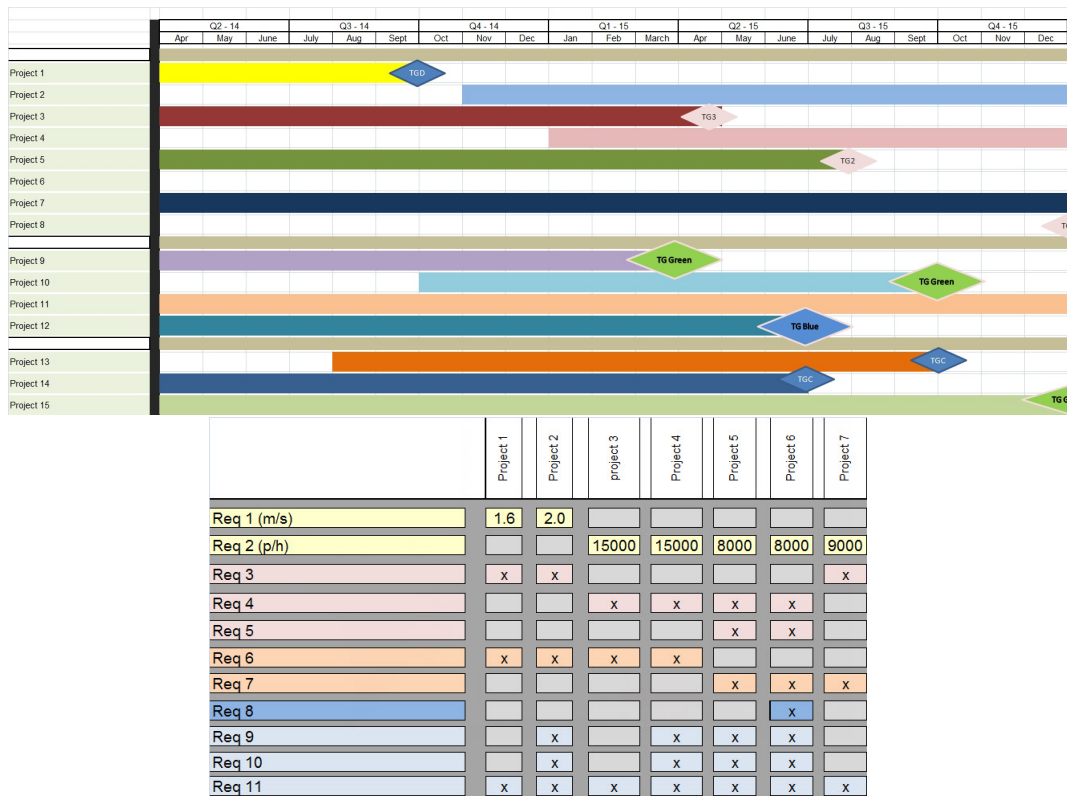


Figure 3.17: Example of Project Roadmap and Requirements vs Projects matrix

The product baseline that consists of a functional breakdown of features that the product provides and a system breakdown structure showing how the system is built (which are the components of the system) must be available. Some other architectural views (Architecture description, AD) could be included in the baseline (Figure 3.18). In addition to the architectural documentation also a System Requirements Specification (SRS) should be provided in order to keep track of the rationale behind the solution decisions.

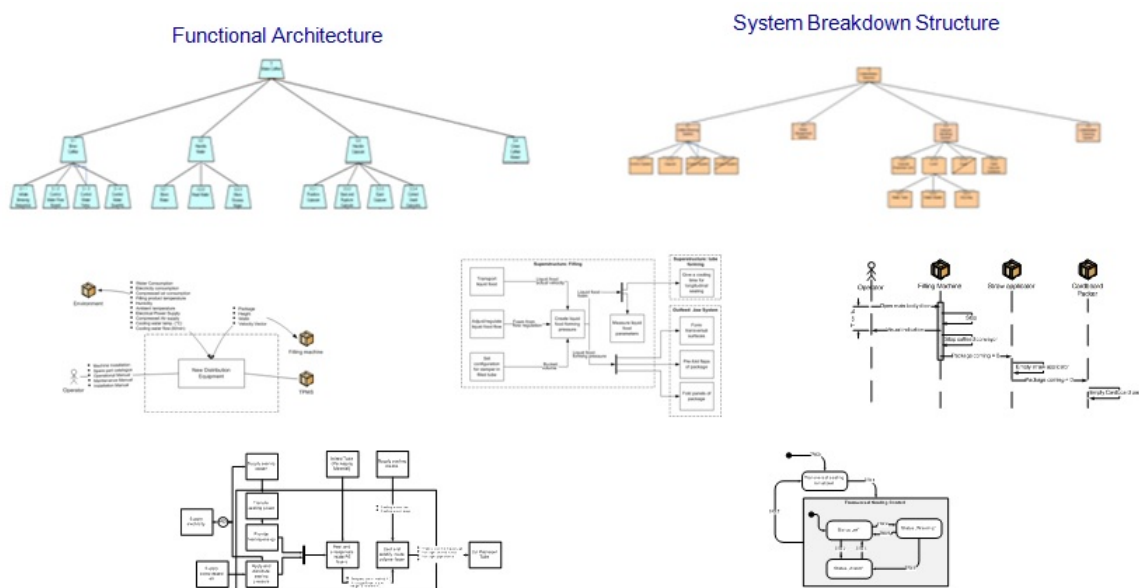


Figure 3.18: Architecture Description Documentation

Requirements vs Projects Review

On the reference baseline architecture an impact analysis of the requirements must be performed. During this activity relevant people (technical experts) must be involved in order to assess implications of product modifications in terms of effort and technical feasibility. The outcomes of the activity are recorded into a Requirements vs Effort matrix that puts in relation requirements to the effort to fulfill them (time, cost and resource).

Architectures Definition

The available architecture baseline is then rearranged in order to group functionalities/-subsystems into three categories (Figure 3.19):

- Product generic are all those features that are needed by the product as a framework, they are in common to all the variants and configuration of a certain product baseline

(e.g. the functionality to fill and seal a package).

- Application generic are the features that are applicable to a defined application of the product, they define the functionalities of the product in a certain application (e.g. features in low cost machines).
- The Specific application features are all the characteristics that are requested to a product to work according a specific need (e.g. type of food to be packed or specific type of package to be produced).

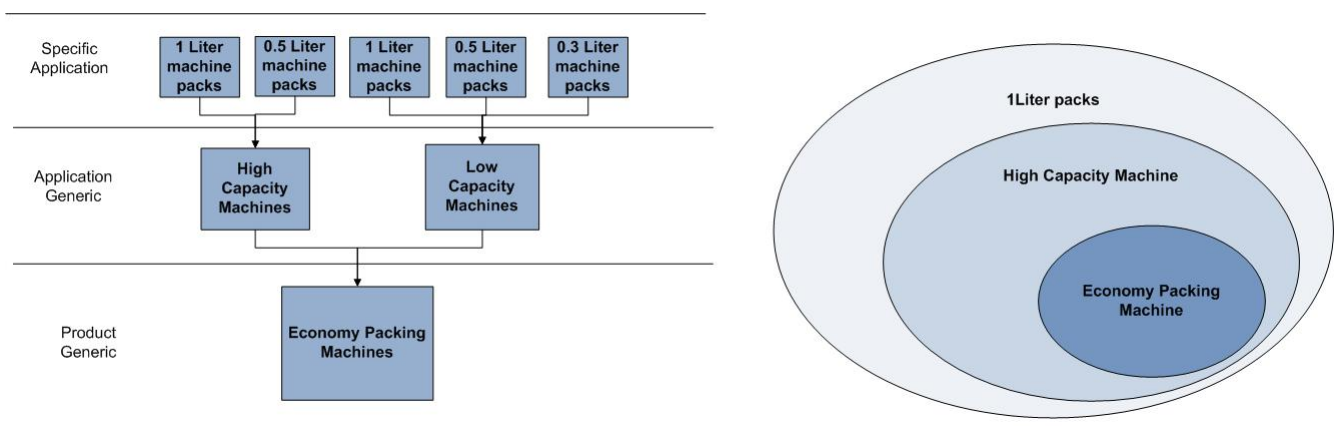


Figure 3.19: Example of Architecture Items categories

The same grouping approach is applied to the new requirements in order to gather the full picture of the new versions features grouped into categories. The outcome of this step is to update a systems vs requirements matrix with consolidated targets. Often a line organization, responsible for the product developments and evolution, could develop a list of additional architectural drivers in order to fulfill the organization and product strategy. The following list shows an example of possible architectural drivers:

- Maximize Commonalities: the more commonalities are present in terms of product components among the different product configurations the better is from a product management point of view (expand as much as possible the Product generic and Application generic part).
- Optimize Development and Management costs: optimization of the resources and costs during the development and life cycle management phases.
- Formalize PLC: a clear picture of the life cycle of the product (i.e. product roadmap, strategy, etc.) enables a faster and effective decision making process.

- Optimize Deliveries: an optimizations of the number of product releases improves the product management process.

Release Versions Definition

At this point of the process the project requirements are allocated to the different architecture categories creating a link with the relative projects. Considering the projects timelines a prioritization of subsystems developments activities must be performed and a map of the subsystems releases must be defined per each identified specific application . All the subsystems releases are then grouped in system releases in order to fulfill the projects deadlines (Figure 3.20). The outcome of this final step is the product roadmap according to projects timeline (Figure 3.21).

Product 1			
	P1v1	P1v2	P1v3
Sub-systems			
SS1	v0	v1	v1
SS2	v0	v0	v1
SS3			v1
SS4	v0	v0	v0
SS5	v0	v1	v1
SS6	v0	v1	v1
SS7	v0	v0	v0
SS8	v0	v0	v1
SS9	v0	v0	v0
SS10	v0	v1	v1
SS11	v0	v1	v1
SS12	v0	v1	v1
SS13	v0	v1	v1
SS14			v0
SS15	v0	v0	v1
SS16	v0	v0	v1

Figure 3.20: Example of Subsystems Versions within the Specific Application n.1 (Product 1)

After some analysis of results of the application of this methodology to real projects[14] a significant improvement in the short term product management is found. It is expected that it affects the long term one as well. The following list represents the main improvements detected:

- Resource optimization
- System complexity reduction
- Simpler product life cycle management

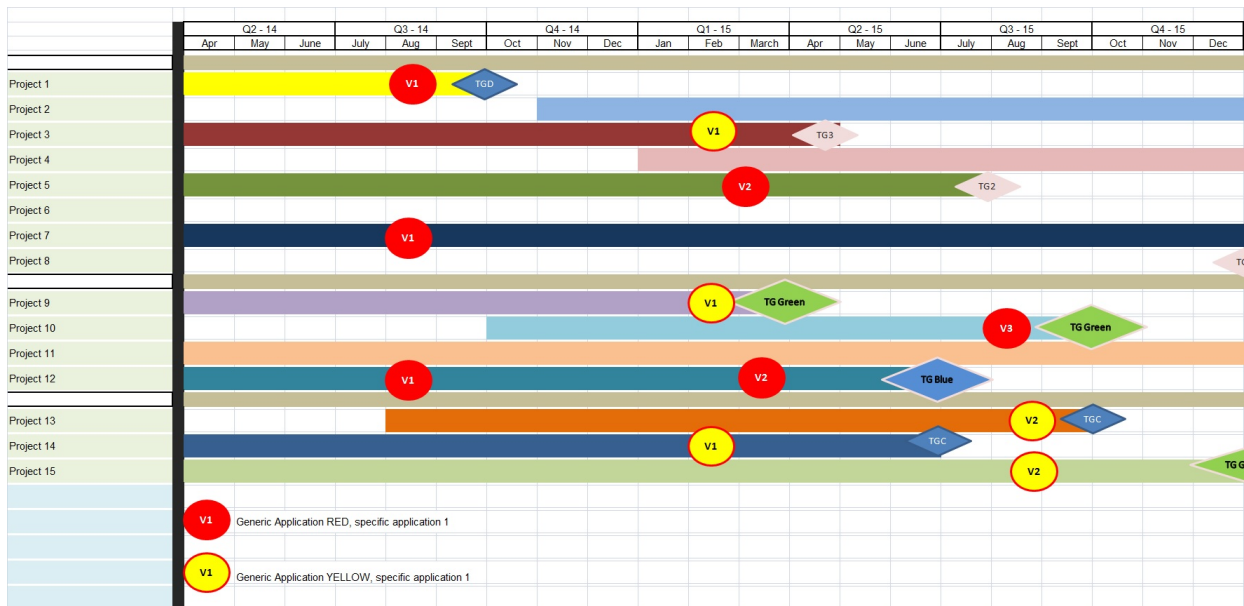


Figure 3.21: Example of resulting Product roadmap

- Improved communication between functional units in the company

While the first three points are largely explained in this section, it is important to spend some words about the last item. It refers to the capability of the system/subsystem team to clearly communicate the development activities to the projects and proactively act in order to fulfill new requirements or changes.

The showed methodology heavily rely on the expertise and skill of the product experts especially in the requirements grouping and prioritization activities. Hence, in order to further improve the described methodology, it should be integrated with structured and formal techniques supporting the above mentioned requirements management activities.

3.5 Discussion

There are many reasons why the addition of services can provide significant growth opportunities, greater stability, and higher profit margins[39]. Among these reasons are:

- Improving predictability of sales and cash flow. Many industries suffer from cyclical variations, e.g., seasonal for the building sector, economic for the automotive sector, etc. In addition, many product categories are becoming more saturated with tough competitors competing for market share. This results in “commoditization” of products and hence lower profitability. Global supply chains, with their

increased purchasing power, are also forcing lower prices, and meeting these demands by improving productivity has nearly run its course. Adding or substituting service revenue can mitigate against some or all of these factors.

- These same pressures on customers, on the other hand, are forcing them to focus on their core competencies and turn to outsourcing to provide many of the functions that were once performed in-house.
- Adding services can help consolidate and protect the core product businesses of a company. Services can differentiate a company from competitors and establish closer relationships with customers. It is relatively easy for a competitor to provide a better and/or lower cost product, but much more difficult to replace an “intimate and trusting” relationship between suppliers and customers.
- Innovation in services typically results in increased customer satisfaction and loyalty. Both are concerned with the direct attributes of the service offering, but also with the image of the supplier, and with the unique relationship that the customer and the supplier may have. Typically, service is about transferring additional values and functions to those gained by just owning or leasing a product to better satisfy customer needs. This will eventually have an impact on the financial results because of repeat purchases by the customer, and because of recommendations to other potential customers.

In this chapter, the implementation of SPD process is shown. The workflow was designed according to the V-Model framework and in analogy with the already existing PD process. In order to facilitate and support some specific phases of the service development process two best practices are described. The first one (Value Chain vs Product Life cycle approach) supports the analysis phase (Discover) by providing a structured way to capture and use the customer and other stakeholder needs. The second one (Product Architecture Management) supports the solution definition phase (Define) by providing an optimizing approach to architecture management driven by project requirements, product roadmap and line organization drivers.

Chapter 4

Case Study and process validation

4.1 Introduction

As already mentioned in Chapter 1 the company strategy included the possibility to expand its own business by integrating services in its own offering in order to affirm its position as full solution provider. One of the enabler of such strategy is the SPD process that has been described in previous chapter. The process has been implemented by taking in consideration the similarities with the existing development processes for the packaging equipment. The benefits of such decision are also a lower impact in the way of working of the organizations and a reduced risk in the activities to be performed. However in order to prove its success the company decided to prove the SPD process by using a pilot project before applying the SPD process to all the service product development projects. The ambition of the company was mainly to validate that the process is appropriate to be applied to a service development process of a certain complexity. Another objective was to educate people in the project team to work with service problematics and thus getting experience with the development of such different products. The selected pilot project aimed to develop a service that, integrating people expertise and software enablers, provided the possibility to improve the diagnostic capabilities of a customer while performing quality checks on its products.

Quality control is one of the most critical activity for food producers and distributors. The activity aims to check if the food processing and packaging process works according to expectations and food industry's standards regulations. The results of quality checks have legal value and must be stored by the customer for further reference. Defects in the food processing and packaging process could have serious impacts on the customer business because it could cause the waste of large amounts of production lots. More critical is the situation of undetected quality issues. In this case, consumers health could be affected. For this reason the impact on business could be seriously high since the

producer must afford costs related to defective product recall and its management. The producer could also be prosecuted by governments or other control organizations because of possible safety issues related to people health. Finally also the producer's reputation is affected with consequent impacts on future consumers trust and sales.

For the above reason and according to the characteristics of service products the service deliverer and its customer must be very close each other. They have to interact a lot in order to secure that a food production system performs well. So if a company wants to deliver a service to customers, its personnel becomes a critical asset for the success [40]. Services, then, require a mix of human action supported by tools and infrastructures to be delivered and then consumed. This highlights the importance of the human interaction into the service provision, thus of the proper "human factor design" when developing service products. The human interaction might be considered as the most characterizing and root of all the other aspects of pure services. As already mentioned above, PSSs are composed by a hard system and a soft system [34], thus as for pure services, they can be considered as services enabled by a physical product. Though they might be considered as standing in between pure products and pure services, even in the Product-Service oriented class of the product-services continuum [46], the characteristics of pure services must be considered as well for the development and operations purposes. That is the human factor has heavy weight and should be taken into account in all the Product-Service Systems development, i.e. it should be designed as well[15].

The following case study description focuses a lot on the aspect of human interaction while following developments according to the SPD process, best practices and Systems Engineering principles. In particular:

- **Requirements collection and management.** The first step is to define requirements to be used in the solution development activities. Human interaction requirements have been developed by using the Use Case Analysis Methodology [7].
- **Architecture design activities.** After the requirement definition activity, the output must be translated in a solution in order to prepare the developments for the service. Typical MBSE techniques [10], [49] and the Product Architecture Management Methodology [14] have been used.
- **Verification activities.** The developed solution must be verified according to requirements expectations. A direct feedback of the customer has been requested before the deployment of the service product according to the principles of producer-customer cooperation described in this introduction.

4.1.1 Outline

The outline of the Chapter is as follows. Section 4.2 provides a description of the above mentioned project steps. In Section 4.3 there are some considerations coming from the comparison between the described design activities and classical design activities for "traditional" products.

4.2 Case Study

A food processing and packaging plant is usually consisting of a certain number of lines that are packing food into packages after its processing. After the food is packed, the packages are staked in pallets and stored in a warehouse. To perform quality controls on products, an operator could use some quality checkpoints next to the production line or bring samples to laboratories Figure 4.1.

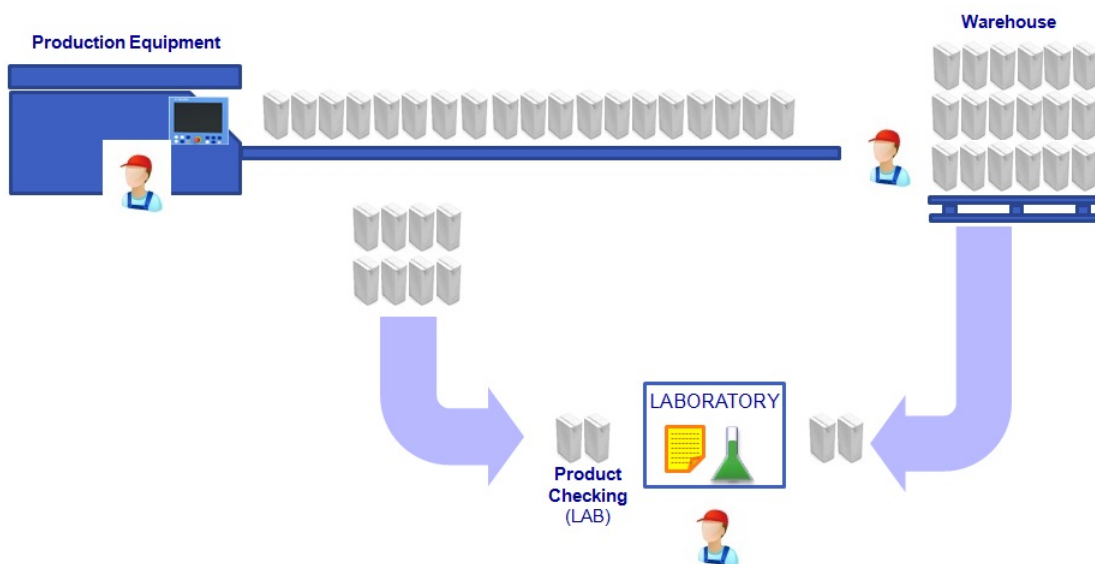


Figure 4.1: Example of production context

An operator dealing with quality checks and operating in the line or in the warehouse need to interact with several tools which the major part is consisting of software. The servicization was meant to be started by adding a software module in the machines they operate to be able to collect and integrate data from various sources. Those data were all about the quality of the machines outcome. The collected data were meant to be used to deliver a quality control service in terms of KPIs, advanced analysis, benchmarking and quality enhancement. These functionalities resulted to be human dependant, thus softsystem of the whole.

4.2.1 Requirements collection and analysis

The design approach is based on the System Engineering principles [6] and SPD guidelines. Thus, the Stakeholder Requirements were collected and analysed first as main project inputs[16]. Afterwards, the whole bundle of high level requirements was translated into System Requirements, reformulating the stakeholders' needs in a technical language suitable for the design activities. In parallel a Reference System was individuated among the company products. The matching with the System Requirements allowed the individuation of the already uncovered features (in this case, most of them were so). In order to develop the requirements related to the human interaction some use cases involving operators were developed (Figure 4.2).

UC#1: Product installation
UC#2: Automatic report provision
UC#3: Warning
UC#4: Advanced analysis
UC#6: Production events data gathering
UC#7: Production checks data gathering
UC#8: Claims data gathering
UC#11: End-sotrage warehouse data gathering
UC#13: Manual entries consistency check
UC#15: Access

Figure 4.2: Example of use cases developed

Each use case consisted on a diagram based on SysML syntax [31] (Figure 4.3).

In combination with the diagrams a detailed description of the use case was developed in order to clarify the use case and improve the information to develop the requirements (Figure 4.4).

4.2.2 Architecture design activities

Though the object of the development is fundamentally different from a classical physical or software product, the activities did not differ in the approach and outcomes. Then, considering the functional architecture and the System Requirements, the Sub-Systems were individuated and the System Requirements were cascaded into Sub-System Requirements accordingly. It is possible to note that, at this point, the Systems Engineering methodologies and the functional product development have a lot in common [35], [6], [24], [41].

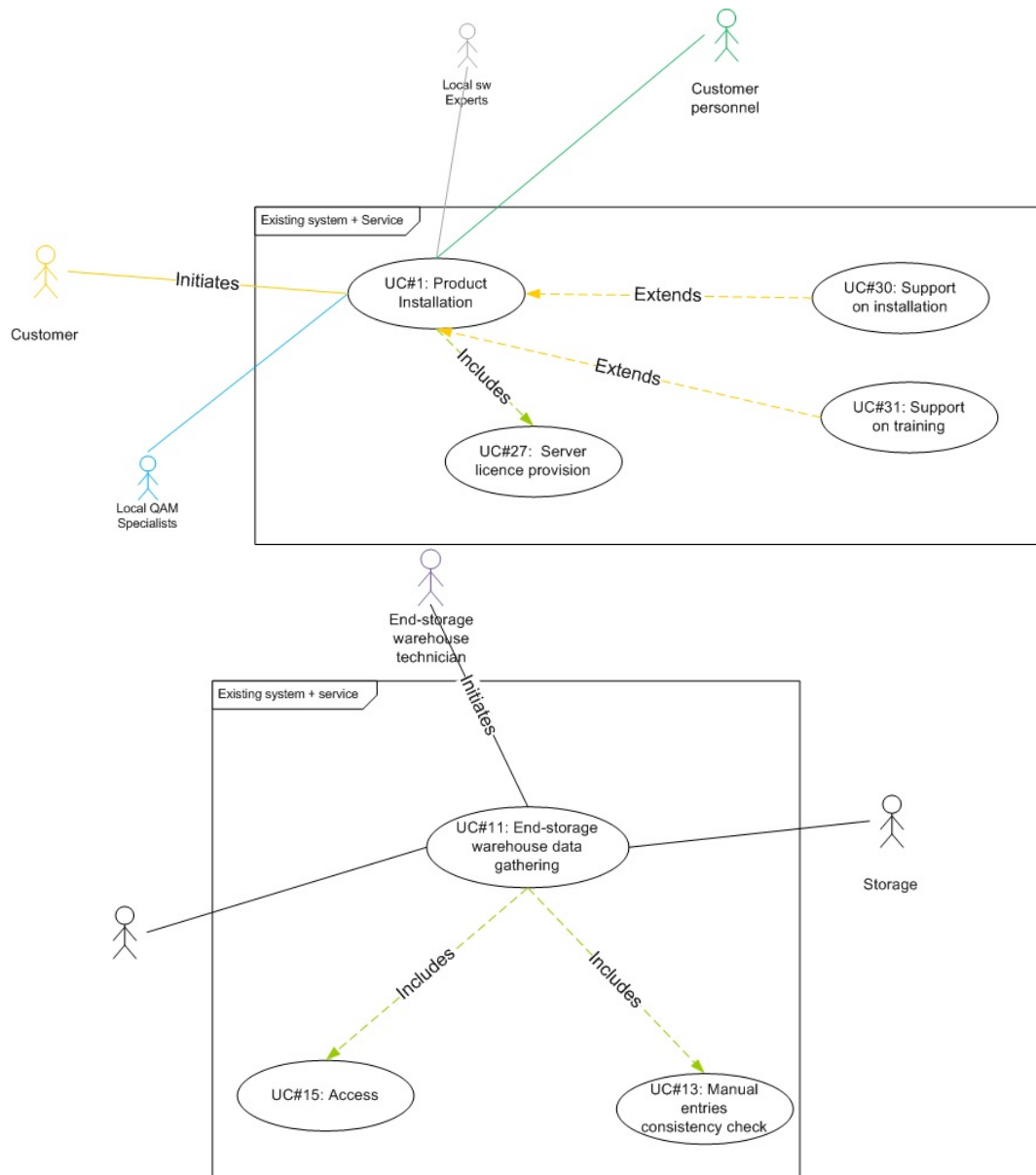


Figure 4.3: Example of use cases involving humans and non human systems

4.2. Case Study

Parent requirement:	
ID: UC#1	Name: Product installation
Author: Service Engineer	Last Updated By: Service Engineer
Date Created: 2014-01-24	Last Modified: 2014-07-15
Current version: 1.2	Owner: Systems Engineer
Operational Modes: XXXX	
External Actors: Customer; Local SW Experts; Local QAM Specialists, customer personnel.	
Initiator: Customer	
Notes: customer personnel: Quality Manager, production operator, laboratory technician, end-storage warehouse technician, claims officer.	
Pre-condition: 1) there is a service agreement between the customer and the service seller; 2) the customer needs to have the Quality software installed on his/her plants.	
Post-condition: the customer has bought the service and has the Quality software installed and ready to be used.	
Normal Flow: In accordance with the service agreement, and considering the indications from Local QAM Specialists, Local SW Experts install and configure the Quality software on the customer machines and terminals. Trigger UC#27: Licence provision. In case of problems extend to UC#30: Support on Installation. Afterwards, Local QAM Specialists provides trainings to the customer personnel. In case of need extend to UC#31: Support on training.	
Alternative Flow:	
Exceptions: XXX	
Assumptions:	
Questions: XXXX	

Parent requirement: SS#0123 End-warehouse related data	
ID: UC#11	Name: End-storage warehouse data gathering
Author: Service Engineer	Last Updated By: Service Engineer
Date Created: 2014-01-27	Last Modified: 2014-07-16
Current version: 1.2	Owner: Systems Engineer
Operational Modes: XXXX	
External Actors: End-storage warehouse technician; Quality software.	
Initiator: End-storage warehouse technician.	
Notes:	
Pre-condition: defects data about the end-storage warehouse have to be inserted.	
Post-condition: there are new record into the Quality SW DB	
Normal Flow: The end-storage warehouse technician triggers the UC#15: Access to access to the Quality software. Then, through a dedicated form, he/she inserts data. The Quality software triggers the UC# 13: Manual entries consistency check and, if it does not return any error, it saves the data into the Quality SW DB.	
Alternative Flow: XXX	
Exceptions: XXX	
Assumptions:	
Questions: XXX	

Figure 4.4: Example of use case descriptions

The first and most important difference with a classical product stepped out already at sub-system level. In particular, they were all due to the human presence: when individuating the Sub-System by considering the wanted features, most of them resulted to be “human sub-systems”, or soft sub-systems [34], indeed. A human sub-system, like the classical sub-system, provides a specific capability to the whole system but they consists of people with their competences, behaviors, defined processes to deliver that capability and tools to support the process accomplishment.

As already mentioned, from a functional point of view there is no difference between a soft sub-system and a classical one, i.e. hard system [24], [41]. Nonetheless, while there is plenty of approaches and methodologies supporting the design of hard systems (or sub-systems) [33], it seems there is a lack of guidance when it comes to the soft system (or sub-system) design.

A capability is composed by people, processes and tools, geared together to perform some specific tasks with certain performance. In a sort of sense, both a system and its sub-systems have to provide one or more functionalities with certain performances. Thus the idea of a sub-system which is a capability is not that far from the usual thought. The design steps for human sub-systems, considering the system engineering principles, were spread at system and sub-system level. The used approach consisted of:

- Sub-system roles and responsibilities definition, started already at system level while identifying the different subsystems, it aimed at specifying and detailing what every sub-system consists of and for what purposes (roles, responsibilities and competences).
- Behavioral schema design, performed entirely at sub-systems level, defined how each sub-system responsibilities interact each other in terms of activities undertaken to deliver the service functions.
- Tool definition and design, performed at sub-system level, aimed at defining and, if required, at designing the tools to be used for the service functions delivery.

These three main steps, which are going to be better detailed below, were reiterated several times in order to reach an acceptable level of knowledge and agreement on the matter.

Roles and responsibilities: who has to perform what

The first undertaken step was to better define the sub-systems identified already at System level. Considering the functions each of them was involved with and the functional

interactions (input from the System Architecture), the picture of who has to perform what was refined. In particular, many sub-systems resulted to match with a role that was meant to be covered by a single person or an organization within the company. The matching was performed in a matrix having the required functions and the sub-systems as dimensions, already at system level. The resulting matrix is shown in Figure 4.5. The blue colored cells identified the functions that were meant to be enabled by the hard sub-system (i.e. the software), while the orange ones identified the functions that were meant to be enabled by human subsystems. It also resulted that to satisfy many functions it would have been necessary to involve more than one subsystem. It pointed out that there would have been the need of internal interfaces between soft sub-systems.

		SUB-SYSTEMS				
		Software	Vendor	Software expert	Analyst	Support
FUNCTIONS	Data Collection	X				
	Data Storage	X				
	Installation & Training		X	X		
	Customization		X	X		X
	Data Analysis		X		X	
	KPIs Provision	X			X	
	Suggestions				X	
	Internal Support					X
	External Support		X	X	X	
	Service Termination		X	X		

Figure 4.5: System Functions vs Sub-System matrix

For confidentiality purposes the data are shown in a simplified and different way than the original one. Then, the information was better refined and cleared by considering the competence availability and competence gaps within sub-systems. Internally available trainings were identified in order to fulfill those gaps. The so created competence map resulted to be very useful to determine the sub-systems involvement into the service delivery activities through all the service product life cycle. Thus all the phases from the

service sale to the end of contract were considered. Several repetition of this step led to further refinements of the roles and responsibilities of each of them (Figure 4.6. In this way the “human ingredient” of the capability was covered.

SS type	SS name	Brief description
Hard System	Software	Module software in charge of collecting quality data
	Vendor	Capability interfacing with customers to develop the most suitable solution. It also works as main interface with the company.
Soft System	Software expert	Capability in charge of managing the module software, its installation, customization and delivering training to users
	Analyst	Capability in charge of performing advanced analysis on data and in providing suggestions on how to enhance the product
	Support function	Capability in charge of supporting all the other ones in their tasks

Figure 4.6: Simplified Sub-Systems roles and responsibility table

Again the displayed information is shown in a simplified way for confidentiality issues. For instance, the service Life-Cycle phases are not displayed into Figure 4.6. The results of this first step affected the whole product architecture and performances. Moreover it also impacted on the company internal structure in terms of involved units, people and roles within them and iterations between organizational units. As already mentioned, the service products have the characteristic of the heavy human importance in terms of service delivery [40]. That means that the human interaction within the service has a crucial importance on the product itself. Moreover it signals that the shift from product to service provider is more than just an enrichment of the product portfolio, but an overall organizational and cultural change [42].

Behavioral design: what has to happen when

Once defined the roles and, afterwards the responsibilities, behavioral schema were drawn for every functions. Sub-Systems roles and responsibilities were the main inputs for this step accomplishment, sided by the sub-system functions matrix. Hence all the information regarding “who is involved with what” were used. Each schema represented, for each function, the activity sequence to be undertaken to deliver it to the customer and/or user. Moreover they clearly depicted which sub-system was responsible for which activity and in which moment of the function delivery. Meaning that the interactions between

4.2. Case Study

sub-systems, and with the external environment were explicated as well (i.e. internal and external interfaces). Behavioral descriptions were provided as well in order to enhance the schema readability. Figure 4.7 provides an example of behavioral schema by showing,

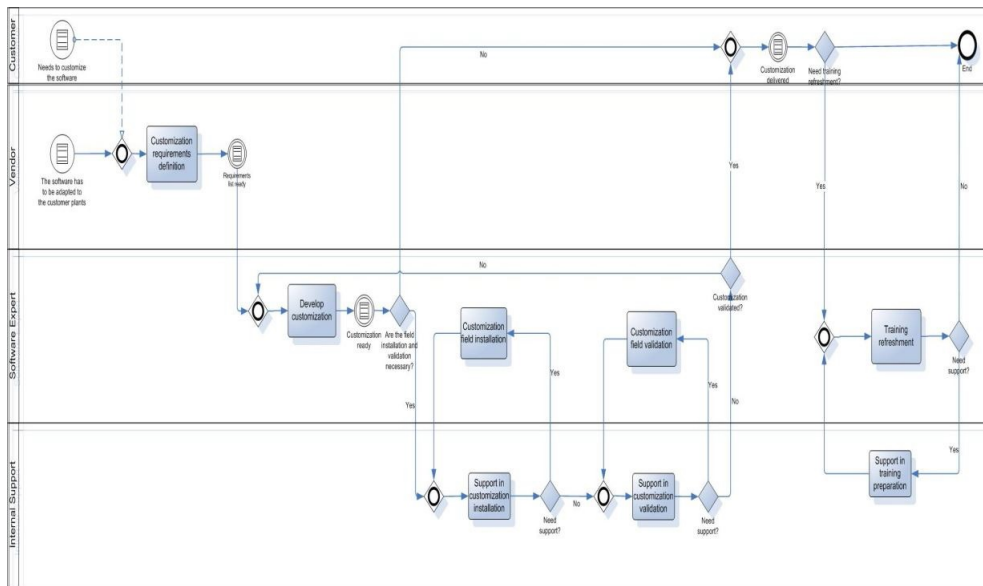


Figure 4.7: Example of diagram schema: "Customization" functionality

in a simplified manner, how the Customization function was meant to be delivered to customer. This function encompassed the software customization on customer request and to adapt it to the customers' installed base. The Vendor, as interface with Customers, was meant to capture the customization needs. Software Experts and Internal Support (if needed) were meant to adapt the software, validate it and to deliver the proper training to customer personnel.

Figure 4.8 displays another simplified example of a behavioral schema. It is about the Analysis functionality. On Customer request, the Vendor provides to Analyst information about the type of analysis required. The Analyst, then, access and retrieves data from the Customer's systems. Then he performs the advanced analysis (being it standard or advanced) and returns the results to the Vendor. Finally the Vendor gives the results to the Customer.

The value of these schema was double because they represented, on one side, the processes of the service product features delivery, and, on the other side, the roles and responsibilities defined in the previous step were translated into a set of activities and tasks. Thus, with this step, the "process ingredient" for the capability creation was covered. As for the first step, this one was repeated several times, with the inputs coming from the previous one. This allowed the knowledge building about the sub-system behavior, a bet-

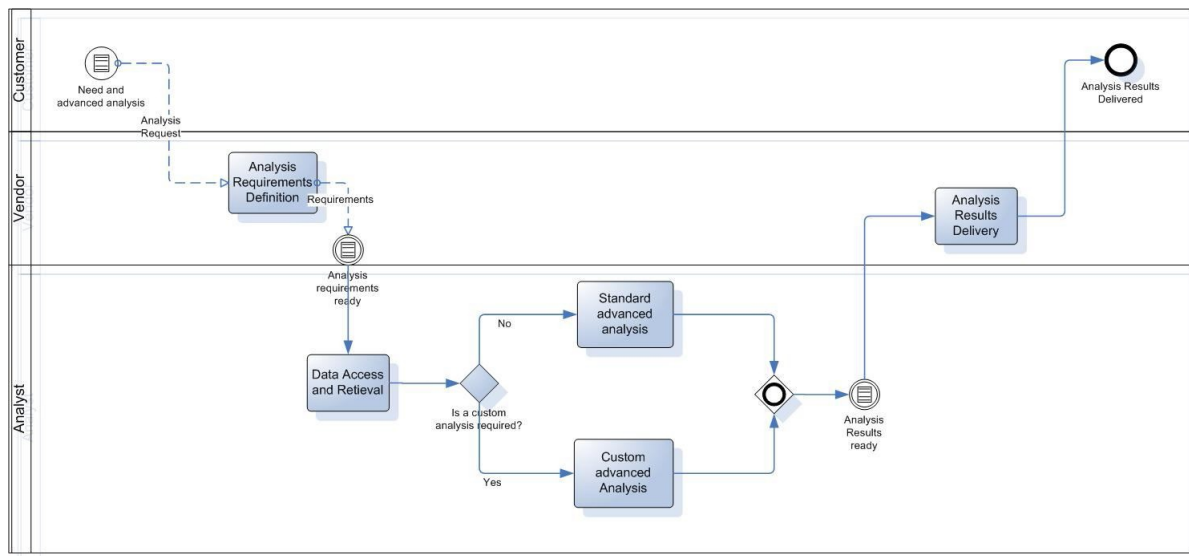


Figure 4.8: Example of diagram schema: “Analysis” functionality

ter definition of each process activities according to the defined roles and responsibilities and it also pushed towards more precise roles and responsibilities descriptions. Moreover, it also allowed a better definition of the interfaces between sub-systems, clearly showing what they have to exchange each other in order to delivery to the customers/users the service features.

Tools definition/design: how it has to happen

Once defined the activities every human sub-system has to accomplish, the tools they have to use for that purpose were defined. That was to allow the proper interactions between sub-systems and with the external environment, in accordance with the defined interfaces. Example of tools may be the data exchange and collection infrastructure, the calculation tools and so on. The required tool list was defined considering their purpose (the activities to be performed) and, of course, the human part. This led to the tools requirements definition to drive their choice or design. Depending on their criticality for the whole service product system, on their internal and external availability and on the required competences for their design, some resulted to be easily provided by suppliers, while others required their own development process. Among the first ones, for example, there was the main data storage management system. Examples of developed tool were the benchmarking one, the calculation methodologies for Analysts and the communication tools instead. In the case of the human sub-systems design, the tools should be considered as enablers for the sub-systems activities. Moreover, it is important to notice that, at

the highest detail level, no “design of the human part” occurred. Only the tools for their activities were object of detailed design activities in the classical meaning of the term. The definition of the tools requirements allowed further discussions and iterations about the behavioral schema and the specific sub-systems roles.

4.2.3 Verification Activity

Another peculiarity of the human sub-system design process comes with their verification activities. The design activities provided a list of sub-systems with their responsibilities, a series of processes and a set of tools they have to use to fulfill the functionalities they are involved with. The highest detail level activity was reached with the tool design. Thus they required verifications activities as well. Those tools did not differ from other classical products. Hence verification activities were carried on as usual through testing on their functionalities and performances. The matter of the human verification still remained anyhow. Object of the design were mainly the interfaces and interaction they should have each other and their role coverage. It is clear that it was not possible to verify them in the usual way as separated systems. The result was that they were verified after their integration as a system, while piloting the service product on equipment. That is because it was necessary to check if the chosen competences were proper, to fulfill eventual gaps, to verify the suitability of the internal interfaces and the use of the tools. For the reason above the need to be effective and efficient during the V&V activities was captured and the project team decided to implement an approach taking inspiration from the AGILE framework displayed in Figure 4.9. The interested reader can learn more about AGILE referring to [3], [8], [20].

In particular the project team had to quickly implement and verify some new requirements coming from the piloting of the service together with the issues and bugs detected on the software during its verification [12]. The AGILE approach with the creation of the backlog and the loops of feature implementation can be very useful in this specific case[20]. The Systems Engineer of the project created the backlog in Figure 4.10 In practice the principle behind the showed picture is that every week a new release of the service was available. After the release a verification activity started to prove the correctness of the solution. in parallel the development activity for the next planned release started. In this way after release n.2 the development team was concentrated in developing the planned functionalities in release n.3 plus fixing the issues detected during V&V activities in release n.1 and n.2. This mechanism was repeated until no more issues and functionalities were to be solved or implemented. the power of this approach is the flexibility of planning for releases and the capability to fix issues while developing functionalities.

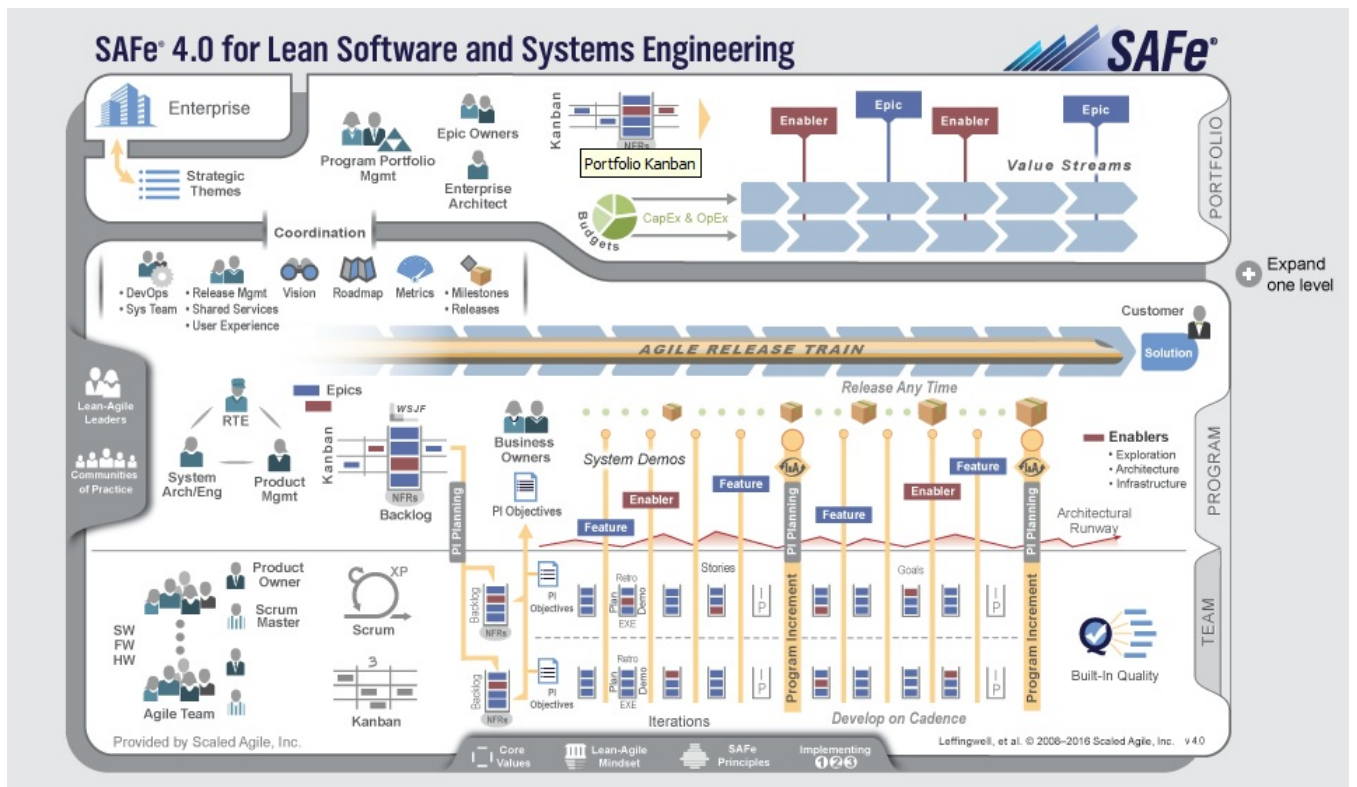


Figure 4.9: Scaled Agile Framework

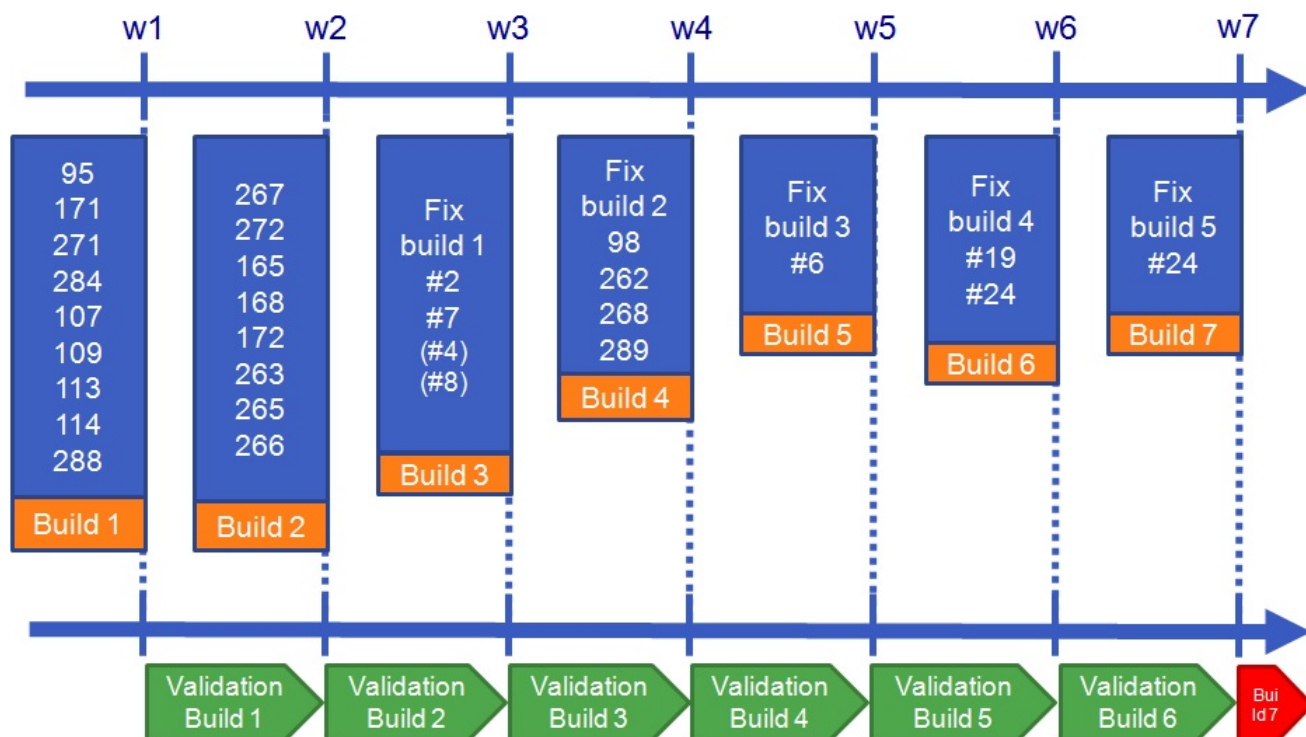


Figure 4.10: Example of Project Backlog according to AGILE principles

4.3 Comparison with classical design activities

The general development approach remains unchanged from the system level to all the following ones: requirements have to be studied, defined and challenged per every sub-system and, according to them, sub-system architectures have to be designed. Anyhow, when people are active part of the (sub) system, they have to be considered as a full capability consisting of competences, processes and tools. Thus, they require those three elements to properly deliver their function. It means that the design should point at defining the required competence profile to cover the specific roles, the processes (i.e. the activities, the interactions and interfaces) they are involved with in order to allow their capability and the tools and infrastructures they should rely on. Anyhow the detail level is not high as it may be reachable when digging from sub-system to module to component to single part design. Nonetheless there might be room for the design of the above mentioned tools if not already available. This activity may be considered even as a product development indeed, with requirements coming from the sub-system and its specific solution architecture. Accordingly to these facts, while it is possible to run verification activities on the developed tools, the approach should be different for human sub-systems. The way chosen by the PSS project was to verify all the subsystems together

while running the whole service-product in a pilot. That was due also to the defined behavioral architectures: in many cases the function delivery involved more than one human sub-system, thus their interactions were fundamental for the verification. Without those interactions, it would have been impossible to verify the features.

4.4 Discussion

In a market environment with an ever growing competition, the shift from a product value based business model towards a product-service integrated value based represents a juicy business opportunity [29]. Considering the substantial differences occurring between the classical products and service products, the servicization of the offering is not trivial. The service business implies several organizational changes, from the culture to the way of doing things [37], [47]. For instance, given the importance of the human component (i.e. the soft system [34]) for the successful service delivery, already existing new product development processes (as well as all the other internal processes) have to be adapted in order to cope with that factor[37]. This is what this Chapter addresses, i.e. an applied approach guiding the design activities for soft sub-systems within a structured framework such as the Systems Engineering principles in a manufacturing company. The proposed methodology was tested in a case study, a product-service system product development project aiming at enriching and integrating the company portfolio with a quality control and enhancement service. The in place product development approach resulted to be fully applicable to the product-service case, with some small tuning though. They were necessary to consider the human factor within sub-systems. The adaptation implies that sub-systems have to be considered as capabilities providing some features and composed by people with their competences, processes to guide the people activities and interrelations, and tools and infrastructure to support them. Moreover the maximum detail level for the design activities was reached during the tools and infrastructure design ones. Due to the human factor, other than defining roles, responsibilities and detailing what they have to do it, the detail level of the design activities is low. The methodology was tested in other projects and in different environment, in order to validate, refine and make it available in the list of best practices for the SPD process.

Chapter 5

Concluding remarks

Food processing and packaging companies face unrelenting pressure from powerful customers and competitors to lower prices and accept shrinking margins on sales. They have responded to this pressure by adopting innovations in operational excellence, e.g., lean manufacturing and six-sigma. As these innovations approach their limits, companies are starting to seek revenue growth from new products and services. They must offer their customers something different than their competitors offer in order to avoid the same low-margin trap that they now face. This thesis suggests that a powerful way for the companies to do this is to offer customers new products and services that allow more efficient and effective use of the products that they currently sell. These new products may complement existing products, and require new manufacturing and design skills, but offering new services is uncharted territory for most companies. Their service experience is often limited to offering customers free or below-cost installation, training, and maintenance. They must learn to offer services that can make their products yield greater total return over their useful life than can a competitors products. These services include customization of products to specific customer uses, training for optimal performance, product disposal, and even taking over customer operations that pertain to the use of the product.

The above mentioned services require companies to form deep and trusting relationships with their customers so that they can co-discover ways to make the best use of their products, and learn new ways to develop and implement ideas for new services. Our literature review led us to propose a “total solution development” model that synthesizes existing new product and new service development models and reflects the dynamic relationship between the company and their stakeholders as well as the complex problem-solving required in such endeavors.

This thesis dealt with the development of a Service development process (SPD) to enable the Service Products production by the food processing and packaging company.

The new process was based on the well known V-Model framework in order to be easily integrated with all the other existing company processes (e.g. PD). Per each phase of the process we identified some supporting best practices. The first one was the Value Chain vs Product Life Cycle approach that put in relation the company value chain with the product life cycle in order to capture in a structured way the stakeholders needs. The approach enabled the extension of existing products and could be very effective to develop and structure the connected service functionalities. The Product Architecture Management methodology was described to support the solution definition phase of the process. The best practice optimized the product architecture when the system has to fulfill more than one concurrent or future project. The output of the methodology was the product roadmap to be used for planning and communicating the product releases and related functionalities.

We also described a case study that implemented the described concepts and methodologies compelling new business models that were accompanied by the use of information technology and proprietary software to help customers use their products more effectively. The project scope was to develop a service that simplified and improved the customer quality control activities. We used the project as test bench to validate the SPD process before full deployment. The success of the project was shown by describing the project phases with deliverables developed according the process. A use case analysis was used to develop human interaction related requirements. The architecture of the product was developed taking in consideration the role and interactions of the so named "human sub-systems" that were treated differently than the other sub-systems. The verification activities for the project were performed with a simplified version of an AGILE approach to development and V&V that allowed the project team to work in parallel with new functionalities development and issue resolution. A final comparison with classical activities in development processes for "traditional" products was also performed and the outcomes led to the consideration that in order to introduce service products in the company business also changes in organization structure and culture are almost always required to do this effectively.

Bibliography

- [1] Gaetano Cutrona Andrea Margini and Cesare Fantuzzi. "structured product development process implementation for a packaging company". *CECIT 2015*, 2015.
- [2] Tan A.R. and McAloone T.C. "characteristics of strategies in product/service-system development". *International Design Conference - DESIGN 2006*, 2006.
- [3] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. *Manifesto for agile software development*, 2001.
- [4] Roy A. Boggs. *The SDLC and Six Sigma. an Essay on Which is Which and Why?* *Issues in Information Systems*, 1(1):36–42, 2004.
- [5] Boston Consulting Group. *Innovation 2006*.
- [6] Dennis M. Buede. *"The Engineering Design of Systems"*. John Wiley & Sons, Inc., v. 2 edition, 2009.
- [7] Alistair Cockburn. *"Writing Effective Use Cases"*. Addison-Wesley, 2001.
- [8] Amaral D. C. Da Silva S. L. Magnanini de Almeida L. F. Conforto E. C., Salum F. "can agile project management be adopted by industries other than software development?". *Project Management Journal*, 45(3):21–34, 2014.
- [9] Gaetano Cutrona. "Analysis of a Structured Development Process for Packaging Machines". Technical report, INCOSE - Italian Chapter Conference on Systems Engineering (CIISE2014) Rome, Italy, 2014.
- [10] Jeff A. Estefan. "Survey of Model-Based Systems Engineering (MBSE) Methodologies, Rev. B". In *International Council on Systems Engineering*, San Diego (CAL), 10, June 2008. INCOSE, INCOSE Technical. Publication, Document No.: INCOSE-TD-2007-003-01.

- [11] EY. "Unwrapping the Packaging Industry". 2013.
- [12] C. Fantuzzi G. Cutrona, A. Margini. "a lean agile approach to service products verification and validation - a case study from a packaging company". *INCOSE - AISE, Verification, Validation and Testing Seminar*, 2016.
- [13] J. Gadrey and F. Gallouj. "*L'innovation dans l'assurance: le cas de l'UAP*". UAP and the French Ministry of Research-University of Lille, 1994.
- [14] Andrea Margini Gaetano Cutrona and Cesare Fantuzzi. "product architecture management - an approach to product life cycle". *INCOSE - CIISE2014*, 1300(3):138–147, 2014.
- [15] Andrea Margini Gaetano Cutrona and Cesare Fantuzzi. "product service system design: How to design humans application of a methodology in a pss development with high human involvement". *Journal of Engineering and Architecture*, 3(2):114–124, 2015.
- [16] Andrea Margini Gaetano Cutrona and Cesare Fantuzzi. "value chain vs life cycle approach for product extensions". *Journal of Engineering and Architecture*, 2016.
- [17] A. Gustafsson and M. D. Johnson. "*Competing in a Service Economy: How to Create a Competitive Advantage Through Service Development and Innovation*". Jossey-Bass., 2003.
- [18] Kasperekand Hollauer Hirth, Maisenbacher and Maurer. "an approach to reveal starting points for pss design support with dynamic models". *CIRP 30*, 2015.
- [19] Mahendra S. Hundal. "*Systematic Mechanical Designing: A Cost and Management Perspective*". ASME, 1997.
- [20] Scaled Agile inc. Scaled agile framework, 2015.
- [21] INCOSE. Systems Engineering Vision 2020. Technical report, International Council on Systems Engineering Seattle, USA, 2007.
- [22] INCOSE. "*Systems Engineering Handbook - A Guide for System Life Cycle, Processes and Activities*". Wiley, v. 4 edition, 2015.
- [23] Project Management Institute. "*A Guide to the Project Management Body of Knowledge: PMBOK Guide*". 2013.

- [24] Röhmbäck A. Isaksson O., Larsson T. C. "development of product-services systems: challenges and opportunities for the manufacturing firm". *Journal of Engineering Design*, 20(4):329–348, 2009.
- [25] ISO/IEC/IEEE. "ISO/IEC/IEEE 42010:2011 - Systems and software engineering, Architecture description". ISO, v. 1 edition, 2011.
- [26] ISO/IEC/IEEE. "ISO/IEC/IEEE 15288:2015 - Systems and software engineering, System life cycle processes". ISO, v. 1 edition, 2015.
- [27] J. Mike Jacka and Paulette J. Keller. "Business Process Mapping Workbook". 2009.
- [28] RGM Kemp, M. Folkeringa, JPJ de Jong, and EFM. Wubben. "Innovation and Firm Performance". 2003.
- [29] Ronnback O. Lindhal M., Sakao T. "business implications of integrated product and service offerings". *CIRP IPS2 Conference*, 2009.
- [30] Hans Loof and Almas Heshmati. On the relationship between innovation and performance: A sensitivity analysis. *Economics of Innovation and New Technology*, 15(4-5):317–344, 2006.
- [31] Object management Group. "OMG Systems Modeling Language". OMG, 2012.
- [32] Collina L. Manzini E. and Evans S. "solution oriented partnership". *Cranfield University*, 2004.
- [33] Zwolinsky P. Maussang N. and Brissaud D. "product-service system design methodology: from the pss architecture design to the products specifications". *Journal of Engineering Design*, 20(4):349–366, 2009.
- [34] Sakao T. Muller P. Towards consolidation on product-service system design. *CIRPS IPS2 Conference*, 2010.
- [35] Morelli N. "product-service systems, a perspective shift for designers: A case study: the design of a telecentre". *Design Studies*, 24(1), 2003.
- [36] Smeal College of Business. "Product and Service Innovation in Small and Medium-Sized Enterprises". United States Department of Commerce, 2006.
- [37] Kallenberg R. Oliva R. "managing the transition from products to services". *International Journal of Service Industry Management*, 14(2):73–99, 2003.

- [38] A. Pysteer and D. Olwell. *"The Guide to the Systems Engineering Body of Knowledge v. 1.1"*. The Trustees of the Stevens Institute of Technology, Hoboken, NJ, 2013.
- [39] W. Reinartz and W. Ulaga. *"Growth beyond the core."*. Financial Times., 2006.
- [40] Abbot S. *"Is it a product or a service?"*. Abbot Research & Consulting, 2010.
- [41] Lindstrom J. Sas D. "advancing development of product-service systems using ideas from functional product development". *CIRP Design Conference*, 2014.
- [42] Sakao T. "integrated product service engineering". *Linköping University, Institute of Technology*, 2011.
- [43] McAlone Tan and Gall. "product/service-system development – an explorative case study in a manufacturing company". *ICED'07*, 2007.
- [44] Giuditta Pezzotta Mario Rapaccini Nicola Saccani Thomas Burger, Walter Ganz. Service development for product services: a maturity model and a field research. *ASAP Service Management Forum*, 2013.
- [45] J. Tidd and F. Hull. *"The organization of new service development in the USA and UK"*. SPRU Science and Technology Policy Research - University of Sussex, 2002.
- [46] A. Tukker. "eight types of product– service system: eight ways to sustainability?". *Wiley InterScience*, 2004.
- [47] Reinartz W. J. Ulaga W. "hybrid offering: How manufacturing firms combine goods and services successfully". *Journal of Marketing*, 2011.
- [48] S. Vasilije Vasić and Lazarević P. Mihailo. "Standard Industrial Guideline for Mechatronic Product Design". *FME Transactions*, 36(3):103–108, 2008.
- [49] VDI. *Design Methodology for Mechatronic Systems (VDI 2206)*. Beuth Verlag, Berlin, 2004.
- [50] Mathias Weske. *"Business Process Management"*. 2009.