POLITECNICO DI TORINO

Department of Mechanical and Aerospace Engineering

Master's Degree in Automotive Engineering

Master's Degree Thesis

Analysis of a hybrid project management approach between Waterfall and Agile. Case study in the European automotive sector.

Supervisor:

Prof. Paolo Chiabert

Company Advisor:

Roberto Cavarero

Candidate:

Giacomo Giavarina

S267892

Academic Year 2020/2021

Abstract

The aim of this paper is to deepen the management approach in an actual case study of implementation of PLM systems for a luxury Automotive European company. This paper is company confidential, the findings and results are reported anonymously, for this reason from this point on the Automotive car company will be addressed as the Client. The implementation of the systems was carried out by Accenture Italia S.p.a., consulting company, exploiting PTC software suite. The first part of the paper is dedicated to the introduction to the different subjects, from the current and future automotive scenario challenges to the definition of the PLM and description of the software suite. Whilst in the latter section the focus is on the management approach with which the project Early BOM was developed. Methodologies, tools, and criticalities will be reported to assess the value of the project management approach applied with a focus on the UAT phase and results.

List of figures

List of tables

Acronyms

ACN	Accenture
API	Application programming interface
APM	Agile project management
BEV	Battery electric vehicle
BOM	Bill of material
CAD	Computer aided design
CAM	Computer aided manufacturing
CR	Change request
DoD	Definition of done
GUI	Graphic user interface
IIoT	Industrial internet of things
IoT	Internet of things
IS	Information system
IT	Information technology
JIT	Just in time
NPD	New product development
OEM	Original equipment manufacturer
PHEV	Plug-in hybrid electric vehicle
PLM	Product lifecycle management
PTC	Parametric Technology Corporation

SOP	Start of production
SUV	Sport utility vehicle
UAT	User acceptance testing
WIP	Work in progress

Table 1 - Acronyms

List of contents

Introduction

The Client requested an overall remodeling of its IT processes related to the management of the product lifecycle focusing in particular on the NPD process. An automotive company largely depends on the management of the data and the generation of different BOMs which allows to organize the development, production, and management of a new and/or already existing vehicles. The so called Ealy BOM project aims at generating a set of applications which align the different BOMs and ease the management and history recording of the data of a product. This remodeling of the IT processes is performed exploiting the PLM software features, PLM stands for Product Lifecycle Management. The aim of PLM is managing the product data and processes from its concept to the end of life. The software used for the Early BOM project belong to the PTC suite: Windchill and ThingWorx Navigate. The latter allows for the generation of a customized GUI to ease the use and the readability of the data which are systematically organized in Windchill. To use correctly a PLM software it is required a minimum level of training; the idea to mask, with a new user interface, the PLM Windchill software, using ThingWorx Navigate, generates from the need of non-trained users to have the same accessibility and ability to work on the system. A specific GUI was implemented for each application requested by the Client, thus transposing their current business processes into the PLM systems.

The project was faced trying to apply the Agile Project Management approach, which granted for a continuous communication with the Client at the end and during each development sprint. The APM approach is based on the idea of pushing the development into a determined amount of time trying to get the most out of it. Due to the difficulties of applying the APM to the PLM, the APM model was modified trying to suit the best the PLM necessities.

An overview on the main classic management approaches is reported to ease the reader in understanding the hybrid solution adopted to conduct the project.
Once defined the project management model applied, a focus on the Test case identification and results on the UAT is reported. These results allow to understand how well the management model applied was able to support the development and to deliver to the customer the business value and the functionalities that satisfy their requirements.

1 Automotive scenario

In this section a brief panoramic on the current Automotive market scenario is reported to introduce the reader to what the Client is facing. The focus of this paragraph is to deepen the current and future challenges that the market is encountering, describing what is perceived as luxury and premium when referring to vehicles and lastly, due to the current pandemic world situation, an overview of the impact of COVID-19 to the automotive sector.

1.1 Luxury and Premiumness

The Client is a European luxury car manufacturer, but what does luxury or premiumness mean in the automotive context? "There is currently no articulation about it, nor have any tools been developed to evaluate luxury during the NPD or manufacturing process" [1]. As stated, there is no articulation between these words and the automotive context, but some considerations are extrapolated from the letteral definition of luxury and premium and their synonymous.
Luxury is a very personal and an internal concept. Luxury is less related with price and more with how people feel. It is a personal feeling, different people have different perception of luxury, and for this reason it is difficult to be described. People belonging to different social classes will explain the concept in different ways. For the low-end social classes, for example, luxury is perceived as economy stability or the possibility to own a car. On the other hand, for the high-end social classes, luxury is not about the economic availability to purchase a car, but about the driving experience that the car provides.

"Exclusivity and uniqueness are fundamental principles of luxury" [2]; this concept is largely used in the luxury automotive field, many luxury and sport brands are used to creating few-off series (vehicle models which are produced in a limited number) or one-off vehicles (usually handcrafted single units on private commission). Customization and personalization are felt by the customer as a way to make their own vehicle unique. Luxury car manufacturers are already used to provide this value allowing customers to select from a long list of optional what they like, also allowing colors/material personalization in a variety of combinations. Luxury brands provide tailoring-like buying experience. This is a strong point that car manufactures must face managing logistics, production planning and product development, challenge that can be tackled using PLM integrated systems.

"A key difference between luxury and premiumness is that the latter is more about product quality and about how the product is marketed" [1], something that can be sensed as the achieving of a certain status, thus on how the buyer thinks to be perceived by others. Premium is about price, that indeed it is a good quality indicator, and a tool used to persuade customer that the higher is the price the higher the quality.

Car manufacturer while managing luxury, providing exclusivity and customization, must ensure to deliver for a price a determined level of perceived quality. Premium is felt by the customer and should not be assessed by engineering data, quality should be the one perceived by customers. Premium feeling is what sets the heritage of brand or of a vehicle line-up, customers are the company best advertising.

1.2 Current and future challenges

The general automotive scenario is facing new challenges [3] that each car manufacturer must address and solve to get a piece of future profitability:

- Complexity and cost pressure: the complexity is increasing to withstand the more and more stringent environmental regulations. With increase in complexity costs will raise reducing the profit margin. Complexity is increased by the need of developing derivatives, serving different segments, based on the same platform for different markets as well as the need to invest on new power train technology without knowing which will prevail in the future.
- Diverging markets: new emerging markets (for example, China) will impact the current market scenario, with new segmentation distribution and business opportunities.
- Digital demands: the world is connected; everything is related to internet. Digital platforms are the primary channel of information for customers (websites, online configurators). For this reason, the trend of online shopping is expected to shift also to cars. Nowadays is already possible to buy certain vehicles online; Tesla, for example, gives the opportunity to place, personalize the order and to process the payment online.
- Shifting industry landscape: "as OEMs seek to develop alternative power train technologies, suppliers will likely provide more of the value-added content per car" [3]. Car manufacturers are moving from manufactures to assemblers, the level of outsourcing has increased and will increase more in the future. Suppliers' management it's strategical, they are gaining contractual power since delivering most of the vehicle value.

The ones listed before are challenges that the global market scenario is facing, for what concern the premium/luxury segment to respond to these challenges more derivatives were created thus expanding the vehicle lineup trying to enter those segments where these companies can still find a market niche. An example is related to the SUVs, following the current trend of this specific segment also premium sports car manufacturer have started producing SUVs occupying then a niche that was still vacant. Brands such as Aston Martin, Bentley and Rolls Royce have released in the

last years their SUVs, they aim at those buyers that want the same level of luxury and premium feeling on a vehicle type born to satisfy other needs (SUVs are best suited for carrying the family around and light off-roading). Luxury brands while struggling to find these market niches must keep untouched their heritage and premium feeling avoiding losing brand image.

1.3 Post-pandemic scenario

The current global pandemic situation directly impacted the automotive market, forcing car manufacturer "to rethink their business models for the near future" [4]. Forecasts indicate a drop both in sales and production to return to pre-Covid levels only in 2024-2025. This drop in sales and production is due to the restriction to contain the pandemic, each country has encountered this drop in correspondence to the beginning of their local lockdowns.

The recovery that the market will face in the following years will be characterized by some new trends. People are more interested in hygiene that translates to the need for a contactless buying experience [4]. Cars are bought on the paper without a proper test drive and through online channels. The city-car segment is going to increase in sales despite the other segments, since people living in big cities want to avoid public transportation. Other trends which seemed to take off in the last years, such as renting cars and car-sharing, have greatly slowed down, people start rethinking about owning a private mean of transport.

On the other hand, the emotional situation of this historic period moved people to think that now it is time to take action to change the world and make a leap forward. There is expectation that the post-pandemic market will be characterized by an increased number of sales for what concern BEVs and PHEVs.

Luxury car brands, such as the Client, were less influenced during the pandemic. Sales of these companies have been strong throughout the lockdowns. They aim at a

customers' niche and are less influenced by the economy of scale. Anyway, these companies should take care about the upcoming trends. Luxury car manufacturers are researching and developing alternative powertrains trying to cope the global trend of electrification. For luxury car manufactures electrification is hard to be faced. Luxury vehicles usually are associated with the idea of speed and power, these vehicles beside quality and uniqueness provide emotional driving experiences. With electrification these driving feelings can be lost, electric vehicles handle differently. Luxury car manufactures are facing a challenge in delivering greener vehicles, whilst delivering the driving experience that the customer is expecting.

2 Product Lifecycle Management

2.1 Overview

"Product Lifecycle Management is recognized as one of the most effective approaches for better, fast, and cheaper product development and management" [5]. "In the modern global economy, companies are facing ever-increasing challenges for short time-to-market to enter into the market early, for reduced time-to-volume to occupy the market quickly, and for decreased time-to-profit to get return from market shortly. Product lifecycle management is recognized as one of the key leading technologies to facilitate companies to overcome these challenges, which will offer companies a new way of rapidly plan, organize, manage, measure, and deliver new products or services much faster, better, and cheaper in an integrated way" [5].
The need to develop a new product management system generates from the urge of tackling new, never faced before, challenges. The business scenario has changed and is going to change more and more in the near future. The manufacturing business model has shifted from make-to-stock to make-to-order and is reaching a point in which the level of customization is so high that the product is going to be designed at the moment of the ordering (design-to-order). The oldest business model is the make-to-stock, goods are produced and stocked waiting for the orders; but with increasing demand for short time to market and short product lifecycle and market volatility the model has shifted to make-to-order, in such a way the customization can be tackled in a more effective way and the risk of stock-out is reduced. With the current market situation this trend of customization and continuous innovation will lead a drastic change in how manufacturing companies were used to work, the future leading business model will be the ones that provide the best customer value for the lowest cost possible, in the shortest time and that are able to react to abrupt changes

(leading business models to satisfy these requirements: engineering-to-order, configure-to-order and design-to-order).

To counteract these new challenges technology supports must adapt to the new scenarios, aiming at reducing product development time, improving revenues reducing costs and enhancing manufacturing capabilities. The technologies that suit the best the problem are the ones of PLM. "PLM provides customers, developers, manufacturers, and suppliers with the most effective means by collaboratively managing business activities throughout entire product lifecycle" [5][6].

The idea behind PLM software is to provide an integration and information flow of all company operations. It merges the already existing software which ease the design, such as CAD and CAM, with the company operations, making sure that what's under developing is aligned between all departments, reducing inefficiencies, enhancing communication, and managing capabilities. "PLM software lets companies consolidate multiple applications systems while leveraging existing legacy investments during their useful lives" [5].

"There are companies that supply software to support the PLM process. That software itself is just a tool and cannot make many contributions if the PLM process is not defined first and understood by its users whom it should contribute to at the end. Setting up PLM within the company is a process and project itself" [5][7].

These last sentences represent the Early BOM project scenario. ACN is the company which is providing to the Client the PLM solutions and its implementation. As it will be explained in the section devoted to the Early BOM project one of the first steps for the implementation of a PLM system is the analysis of the so called "AS IS". A review on how the current operations and workflows are managed in the company before the implementation of the PLM system itself. The deployment of a PLM system is a project and a challenge itself.

2.2 PTC technologies

Figure 2.1 – PTC logo

In the Early BOM project the Product Lifecycle Management software suite used is the one of the Parametric Technology Corporation company. PTC is a software and services company that, together with a partner ecosystem, empowers digital transformation for enterprise companies. PTC headquarter is in Boston, MA. The company mission is the one to *unlock the value created by the convergence of the physical and digital worlds*.

2.2.1 Windchill

Windchill is the PLM software of the PTC suite. In the case study described, the Client was already using Windchill as a data management software, solution that does not unleash the real potential of a PLM software. The core of PLM is managing, beside product data, company processes and workflows. Windchill main features are managing MCAD (Manufacturing Computer Aided Design) and ECAD (Engineering Computer Aided Design) and related data, documents, and processes. The system allows to enhance project management best practices connecting deliverables to product data [8].

For what concern the Early BOM project, Windchill represents the backbone of the system of applications that are developed in ThingWorx Navigate. As stated in the name of the project, through the use of Windchill is possible to generate, manage and centralize product Bill Of Material since the early stages of development; an integrated system of product data, company operations and best practices. Each application developed in ThingWorx Navigate refers to data and objects created ad-hoc in Windchill.
The use of PLM software allows for easy tracking of components, thus enhancing the reusability and avoiding proliferation of duplicates also simplifying the suppliers' management keeping track of all history data of already concluded projects; as well as addressing quality issues, providing helpful data for managing, prevent and predict problems.

2.2.2 ThingWorx and ThingWorx Navigate

ThingWorx is an IIoT platform, suited for smart connected manufacturing and smart connected products providing the tools for the organizations to source, contextualize, synthetize and manage their data through various workflows while providing easy desktop and mobile engagement methods. Popular use includes assets monitoring, predictive maintenance, and assets utilization for smart manufacturing. This platform enables the digitalization of IoT information coming from the physical world, also maintaining a high level of flexibility for what concerns the deployment methods. This platform indeed is an end-to-end industrial innovation system that allows for the rapid generation of industrial apps [8]. The ThingWorx platform provides to the OEMs readymade apps such as ThingWorx Navigate, ThingWorx Control Advisor and ThingWorx Asset Advisor.
In the Early BOM project the app utilized is ThingWorx Navigate. This PTC software allows for the generation of industrial apps with concurrent development

without the need of programming. The deployment of the apps is eased using already made widget that can directly interact, thanks to a specific API, with Windchill. In the developing of the system implemented for the Client there was the need to develop ad-hoc solutions with custom widgets. To develop a custom widget the block programming logic with which is possible to program in ThingWorx Navigate is not enough, actual programming skills and knowledge are required; there is the need to write code to customize the widgets. The concurrent development achievable with this platform allows more than one developer to work on the same project on a cloud-based solution.

3 Early BOM

In this section a brief introduction on what is a bill of material is provided, that is also preparatory to introduce the Early BOM project objective.

3.1 Bill Of Material

"The bill of material constitutes the heart of many current information systems for production/inventory control. The bill of material of a particular product is a document that specifies how this product, the parent, is built-up from its immediate components. These components may have a bill of material of their own, and so on" [9]. In the BOM the components are organized hierarchically, accordingly to the product structure. The products at the top of the structure can be considered as the final products that do not belong to any other product BOM, this mean that usually they are those end items that are going to be marketed and sold. On the other hand, the primary products are those ones that do not have a BOM since their simplicity and usually they are directly purchased from the suppliers, they represent the standard components such as, for example, nuts and bolts. The subassemblies are those components located in between the two just cited. They belong to a BOM and at the same time have their own BOM.

At each link of the network the number of components required is reported on the relationship and it is called "quantity per". At each level, beside the number of components, information about the assembling order is reported too to ease the understanding of the structure of the product and its assembly procedures.

This document can be represented in different manners:

- Part list: it is a single level representation; it is a list of the components present in a product.
- Indented BOM: it is a multilevel representation of the bill of material; as the name states it is an indented list, the assemblies are represented at a higher level, the lower the level the simpler the component.
- Assembly chart: it is a multilevel graphical representation of the bill of material.

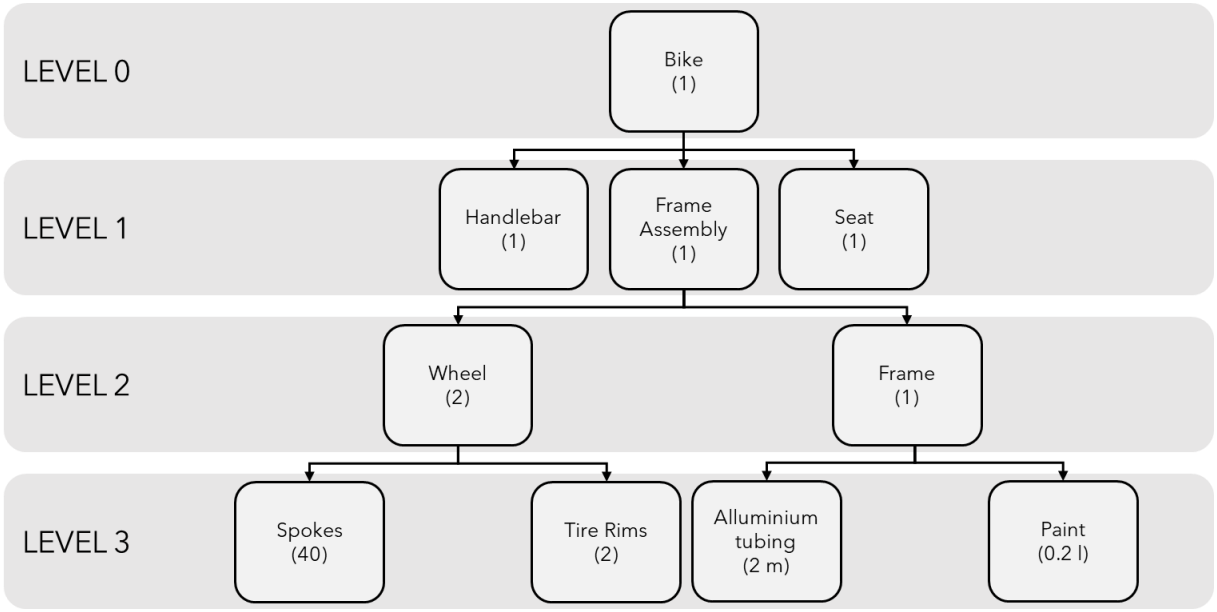

Figure 3.1 – Assembly chart structure bike simple example

There are different types of BOMs, each one satisfies a different need. In the following are reported details on the two most commonly used. The MBOM (Manufacturing Bill Of Material) contains the information about all the components to assemble a product and those information are used to plan the purchasing of the materials and components themselves. The EBOM (Engineering Bill of Material) is a technical BOM which is used during the development process by the engineers, and it is

composed by the set of CAD drawings of the components, assemblies, and final product.

3.2 Project objective

The Client is engaged in a digital transformation program that aims to streamline the NPD fostering the digital product traceability and improving the overall process and efficiency. To achieve this digital transformation the Client is going through a redesign of its IT systems, with the aim to implement a new infrastructure which provides better support for the product structure management process, enabling effective information flow, ensuring the BOM alignment among the different IT tools and improving product data accessibility.

The Client wants to improve its NPD centralizing in the Early BOM the BOM structure of the vehicle, even though the new vehicle still does not exist. The Early BOM at the beginning of the NPD is a high level BOM of the vehicle, indeed it does not contain information on the exact components that will be mounted in the vehicle. This logical product structure is subsequentially filled with components that are design ad hoc or carried over from past projects. This high-level structure allows to centralize the BOMs from the initial phase of development to the SOP.

What the Client is trying to achieve is the implementation of a Generic BOM. The Generic BOM is best suited to avoid redundancies when managing the variants for a product [9]. The structure of the BOM is the same for all the variants and by managing some attributes is possible to adapt the generic structure to the right variant. This process allows to reduce the number of BOM managed by a company and ensures that the BOMs for a product are aligned between the variants of the product itself. Early BOM represent then the structure of the generic BOM for a generic car. It is predisposed at accepting the product specification for the product "vehicle" taking into considerations all its possible variants. Following the advantages

stated above this structure allows to speed up the NPD process removing the need in generating and managing for each new model a dedicated BOM starting from scratch. The implementation of the Early BOM in the IT infrastructure of the Client, leads to the generation of links between the elements in the BOM and the data already present in the other IT systems of the company. These links ease the retrieval of the information reducing the effort to manually keep updated and aligned the departments whilst ensuring to avoid redundancies and data proliferation.

Currently the BOM alignment processes and the structure of the generic BOM are managed by means of an Excel spreadsheet. Excel is a powerful and useful tool, but it is not the right tool for the job. Everything should be updated by hand, trying to educate spreadsheet users to modify only those cells that belongs to them ensuring they do not modify the structure of the Excel to trace all the modifications.

To improve its IT processes, the Client hired ACN to implement the new redesigned PLM solution. This remodeling of the IT processes is performed exploiting the PLM software features. The Early BOM project aims at developing a set of applications that ease the interaction of the users with the PLM integrated system (Windchill). Each one of the applications developed in ThingWorx Navigate communicate to the back-end PLM software by means of a dedicated API. The developed applications are designed to simplify and integrate already existing processes with Windchill, ensuring the correct data recording and alignment, thus easing the NPD process managing the above mentioned Early BOM.

4 Management approach

This section is dedicated to the description of the actual project management approach applied to develop the Early BOM project. The approach applied is a hybrid solution which mergers Waterfall and Agile methodologies exploiting the core values of Lean Thinking.

4.1 Lean thinking

Lean thinking roots originates in Japan, more precisely in the Toyota Motor Corporation shopfloors. Lean thinking owes its name to the 90's book "The Machine That Changed the World: The Story of Lean Production", this book tells about the evolution from car craft production to mass production to lean production [10].
Lean Manufacturing: "Toyota introduced some new innovations to cope with the intense domestic competition and scarcity of resources, such as JIT production system, Kanban method of pull production, respect for employees and high level of employee problem solving/automated mistake proofing" [11]. Lean manufacturing is based on the idea of continuous improvement (Kaizen) to reduce at its maximum the amount of non-value-added activities and waste. In the Japanese literature these concepts can be resumed into three main categories: Muri, Mura and Muda [12].
Muri means overburden, the amount of work requested to the system/people in the plant is beyond their capability. Lean focuses on minimizing the level of stress on the machines and effort required to the worker, this leads to the reduction of employees' absenteeism, illness, and machine breakdowns.
Mura mean unevenness or non-uniformity; it represents the variability of the demand and of the production which consequently leads to the generation of one of the seven waste of Muda.

Muda means waste, in Lean thinking seven types of waste are identified, anything that do not add value to the product or to the customer is a waste.

Seven wastes classification [13]:

- Over-Production: producing more than what is necessary. The quantity produced are independent from customer orders.
- Waiting: it is an action which is not strictly necessary in the product manufacturing routing.
- Transportation: the movement of goods from an area to another generates a waste since transport has a cost.
- Over-Processing: represents all those inefficiencies that cause delays, queues, cost increase and outcome variability.
- Inventory: it is the unnecessary stock of material, regardless of its processing status (raw material, WIP and finished products).
- Motion: waste represented by the unnecessary movement of the worker or of the material during the production process, the motion of the worker in the workstation is a waste of time and an action that does not add value to the product.
- Defects: waste of material, time and increase in cost due to defective parts that need to be reworked or be discarded.

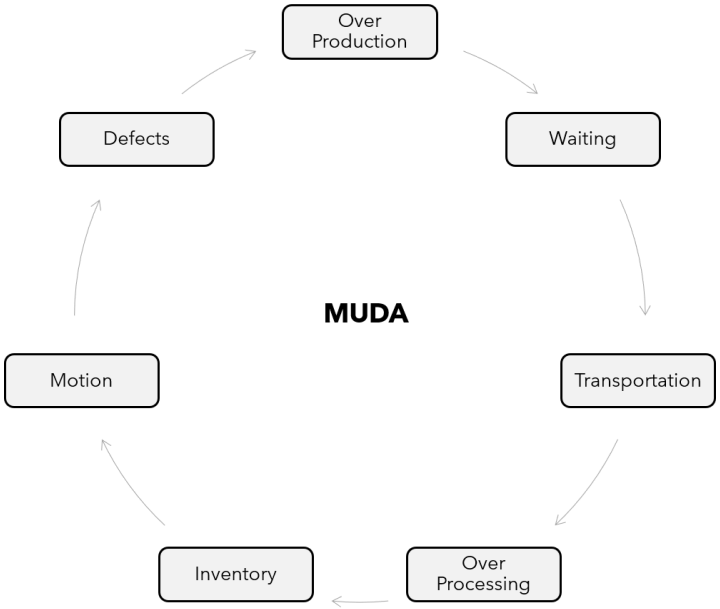

Figure 4.1 – Seven waste classification

We refer to the first implementation of the Lean concepts and values to the car industry as Lean manufacturing; Lean practices in this sector are aimed to the shopfloor. "Lean had moved away from merely "shopfloor-focus" on waste and cost reduction, to an approach that contingently sought to enhance value (or perceived value) to customer by adding product or service features and/or removing wasteful activities" [11]. The change of focus from the shopfloor to the idea of applying Lean values to other sectors enshrines the concept of Lean Thinking.

4.2 Waterfall project management

4.2.1 Overview

The Waterfall management approach is based on the linear sequence of phases, each one of these phases is dedicated to specific actions and cannot starts till the previous one end. It is best suited for those sectors in which the requirements are assessed at the beginning of the project due to the complexity in implementing future

modifications: this approach arose in the construction field where the misunderstanding of the initial requirements is hardly addressable for modifications. The idea behind this model is the one that describes the waterfall reaction of events, one event happens only once the previous one ended, and hence its name. This methodology was formally cited for the first time in an article in the 1970 by W. Royce [14], even though he does not refer to it as the "Waterfall". In the paper the description of this methodology is reported to highlight its criticalities and to suggest a solution to improve it (Royce suggested 5 steps to reduce the development risks), but at that time the imperfect solution became mainstream due to its simple linear sequence of steps [15].

4.2.2 Waterfall phases

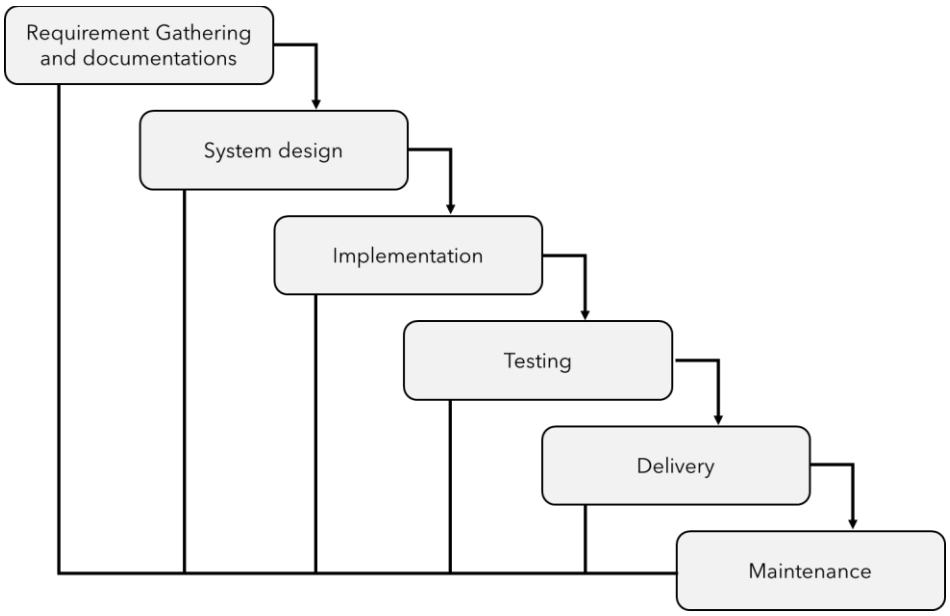

Figure 4.2 – Waterfall phases [16]

4.2.2.1 Requirement gathering and documentation

This phase represents the core value of the Waterfall management approach, as stated in Section 4.2.1, this model is used on those projects in which it is really hard or almost impossible to make any modifications. Thus, the first requirements gathering is fundamental to obtain a successful outcome. Different techniques are used, from questionaries to group brainstorming. Analysis sessions are held usually between the project team and the clients. These analyses are fundamental to ensure that what is going to be delivered to the client comply with the expected result. Secondly once gathered the data, the requirements themselves are specified by generating documents. These documents report the results of the analyses above mentioned.

4.2.2.2 System design

The project team goes through the data collected and define which are the specification of each requirement and thus define which are the tools and methodologies that are going to be applied. To understand which tools are necessary to satisfy the requirements the project team reviews the documentation gathered in the previous phase, this allows for the identification of any critical issue. Since the critical issues are addressed from the beginning of the project, the development team can tackle as soon as possible those obstacles that might cause delays or quality problems ensuring to arrive to the project deadline with a successful product to be delivered. The outcome of the system design phase is the blueprint of the project, it describes how the project team is going to develop the solution identified and the product features.

4.2.2.3 Implementation

Following the project blueprint the development starts. In this phase, if we consider software development project, developers start writing the code or, if we consider a product development project, the first physical outputs are generated, physical products/prototypes are manufactured. Usually, the project team is divided in smaller groups. Each group focus on a particular unit that should be developed and to its testing, this procedure enhances the identification of problems [17]. Each team moves to the following unit only once the previous one tests successfully. These units are subsets of the project requirements.
The development groups at the end of each development carry out the unit testing; these tests focus on the functionalities of each element without considering their integration/interaction with the other units of the system, the process of integration and testing is carried out in the following phase [16].

4.2.2.4 Testing

The single units developed and tested in the previous phase are integrated in the system. The first purpose of the testing is to ensure that the units' interaction and integration is working correctly. Secondly, key user might participate to the test activities to validate the results of the project, the UAT (User Acceptance Testing) usually applied strictly to software development projects. Key users can immediately report feedback and notify if something is not working as stated in the requirements. Prior the UAT the project teams in collaboration with the client team generates the AC (Acceptance Criteria) [18]; starting from the requirements, the criteria are identified. The AC is the set of rules and features that a component or a system must respect to satisfy the requirements. The AC gives an idea of the level of development reached at the time of testing. Their definition is crucial, if the AC is too generic it

covers few functional requisites and thus it does not precisely represent the requirements it is intended to satisfy.

4.2.2.5 Delivery

Once the testing phase ended and the key users accept the product, the delivery phase starts. If we refer to a product development project, in this phase the tested product is put into production and commercialized, on the other hand if we refer to software development, the developed software is integrated into the client environment.

4.2.2.6 Maintenance

For what concern software development projects: once the solution is deployed in the client environment, patches and updates are released periodically to ensure to keep up to date and running the new developed software with respect to the other application already running in the environment. "As problems are found due to improper requirements determination or other mistake in the design process, or due to changes in the users' requirements, changes are made to the system during this phase" [14].
On the other hand, for product development projects maintenance is about the management of the lifecycle of the product, if we refer to a car, for example, it is the management of the production, the management of the marketing campaign, the generation of new restyled models to keep it marketable and its final retirement from the market.

4.3 Agile project management

"On February 11-13, 2001, at The Lodge at Snowbird ski resort in the Wasatch mountains of Utah, seventeen people met to talk, ski, relax, and try to find common ground—and of course, to eat. What emerged was the Agile 'Software Development' Manifesto. Representatives from Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming, and others sympathetic to the need for an alternative to documentation driven, heavyweight software development processes convened" [19]. Fourteen out of these seventeen people became the first signatories of the Manifesto of the Agile Software development.

Manifesto for Agile software development

"We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:
Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan
That is, while there is value in the items on
the right, we value the items on the left more" [20].

4.3.1 Overview

"In recent years, a new and disruptive innovation environment has challenged NPD theory and practice with the emergence of, for example, digital creative industries, co-creation, 3D printing, fast prototyping and the demand for radical innovation-

oriented capabilities" [21]. Companies are facing these new challenges whilst trying to deliver the best possible value for the customers, reducing as much as possible the operating costs and time to deliver results, basically doing more with less. "To face these challenges there is the need of an NPD framework that combine simplicity, velocity, and flexibility as never before" [21].

Traditional approach to the project management is the Waterfall model. Distinct project stages are arranged linearly and only at the end of one stage is possible to move to the next one, following a path set at the beginning of the project and delivering the results within a final deadline. This approach is not suitable to react to the rapid change in the requirements which might be suggested by the Client during the development. There is then the need for a methodology and mindset that can react proactively to any request, delivering in the shortest time the best value: the Agile Project Management [22].

4.3.2 Agile framework

"Agile methods are focused on flexibility by using minimal set of rules eliminating activities that do not add value to the product development process. They are based on series of iterative development cycles and promote self-management and self-discipline attitudes in order to help the team be more responsive to changes" [21]. At the end of each iteration the results are criticized and reviewed by the project team. The output of these meetings is the one to understand what it is required for the following iteration. The main Agile approach advantage is the possibility to react to and solve any issue raised by the client throughout the project at any time, which is fundamental to be able to deliver a successful project on time and within the budget. The first applications of the Agile project management arose in the software development field, where requirements changed too fast, making it impossible to keep up using the standard approach (Waterfall).

In the following sections two Agile frameworks are reported.

4.3.2.1 Scrum

Scrum is a framework for the Agile Project Management based on fixed-length time iterations. The basic concepts of Scrum originate from Lean Thinking and Empiricism. The former, as seen in Section 4.1, aims at reducing wastes focusing on what is essential. The latter establish that knowledge comes from experience and from what is observed.

There are three main figures involved in the process: the Product Owner, the Scrum Master and the Developers.

The Product Owner is the end user, the stakeholder or simply an expert on the project, which is under development. It is a single person and not a committee or a team. He has the duty to generate the product backlog, that is one of the Scrum artifacts, that represents the main expected product features and characteristics. His role is the one to clearly communicate to the development team the product goal, ensuring the product backlog understandability and visibility by means of the management and ordering of the product backlog items.

"Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint" [23]. They are responsible for the identification of those activities to be moved from the product backlog to the sprint backlog. The sprint backlog, Scrum artifact, is a subset of the product backlog, that is filled up till the maximum capacity for the following sprint is reached. It represents the activities which are going to be implemented in the next sprint.

"The Scrum Master is accountable for the Scrum Team's effectiveness. They do this by enabling the Scrum Team to improve its practices, within the Scrum framework. They are true leaders who serve the Scrum Team and the larger organization" [23].

The Scrum Master must ensure that the objective of each single sprint is in target and completed, managing the process, and solving problems to ensure that the Product Owner can drive the development with the development team without hurdles. He fosters these activities by helping the Product Owner defining an effective product goal, and the definition of the product backlog. The Scrum Master is the Scrum theory spokesman across different Scrum teams and the company organization. In the organization the Scrum Master manages and leads the training of Scrum ensuring that this practice is adopted in the correct manner.
The scrum framework is completed by five ceremonies: Sprint planning, Daily scrum, Sprint review, Retrospective, and Backlog refinement.

- Sprint planning: "The whole Scrum Team then collaborates to define a Sprint Goal that communicates why the Sprint is valuable to stakeholders" [23]. The Product Owner and the Developers discuss and identify what are the activities in scope for the sprint, thus those items that are moved from the product backlog to the sprint backlog.
- Daily scrum: also known as Daily stand-ups since held while standing up to reduce the time spent in this activity. The daily scrum is a fast meeting held each day by the developers to align each other on what's has been already completed and what's to do next. These meeting are held to adapt and plan the upcoming planned activities coming from the sprint backlog.
- Sprint review: team meeting where everyone shares with the team what was delivered in the sprint, moreover the increments are detailed. This activity is centered on the inspection of the completed activities, to these meetings stakeholders and key users might participate to suggest what to do next based on what was completed in the sprint.
- Retrospective: analysis on what went well and what didn't, planning to improve the quality and the effectiveness in the next sprint. This activity is performed by the Scrum team trying to identify what were those processes

and changes to be implemented to improve how the team carries out the sprint.

- Backlog refinement: review of the product backlog and subsequent update of its content based on the activity of the running sprint. During the sprint the Developers might encounter some issues and thus there is the need to trace, in the product backlog, the items to be implemented to counteract the problem. These new generated items will be faced in future sprints once moved from the product backlog to the sprint one.

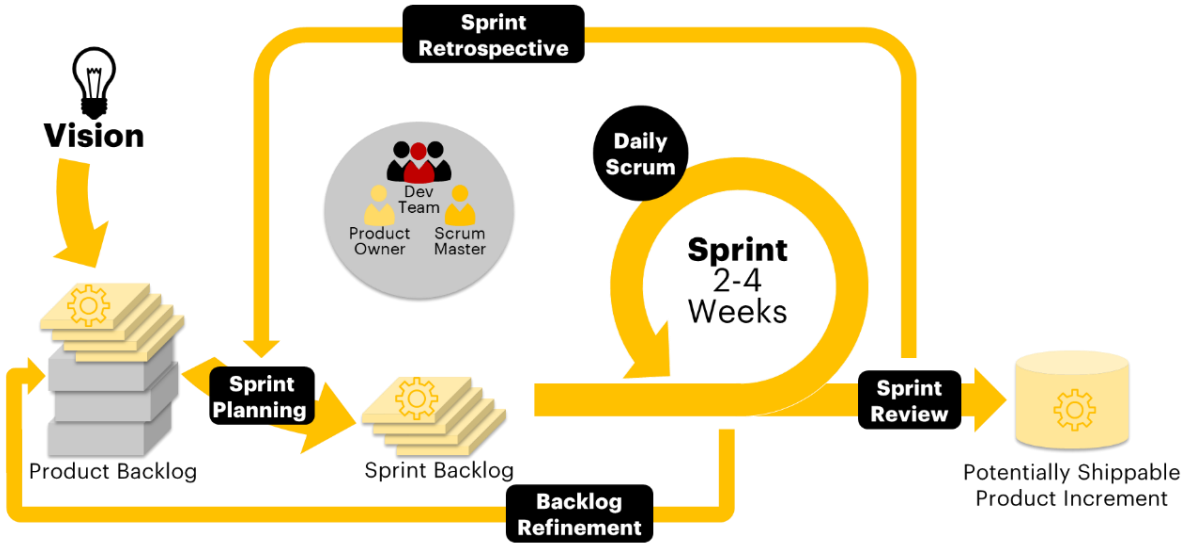

Figure 4.3 – Scrum structure [24]

Scrum framework is usually implemented using a scrum board. During the sprint planning the developers identify those cards to be moved from the product backlog to the sprint backlog. Usually, the board is divided into multiple steps such as “To do”, “WIP” and “Done”. Activities are summarized in cards, each card is assigned to one or more team members and to each card usually has an expiration date that corresponds to the end date of the sprint. It is important to remind, to avoid confusion with the Kanban board explained in the next section, that the number of

cards moved from the product backlog to the sprint backlog is based on the experience of the Developers and Scrum Master. Story points are assigned to each card, the story points represent the forecasted time that the developer needs to complete the activity. The number of story points to be completed are decided in a meeting between the developers and therefore the sum of the story points of the cards moved to the sprint backlog should not overcome the declared story points for the sprint. The amount of work moved to the sprint backlog might overcome the team capacity if the story points assigned to the cards are misestimated.

If an activity is over estimated and more story points are assigned to the card, the team capacity for the sprint is not fully saturated. On the other hand, if the activity is underestimated more time will be required to perform the activity and the team capacity will be overcome.

The developers move a card from the "WIP" column of the board to the "Done" only once the Definition of Done (DoD) is completely satisfied. The DoD is a set of actions and rules defined by the Scrum Team to define that an activity is totally completed and ready to be delivered. It is a checklist, all the voices must be completed before declaring a card as "Done". Once an item of the product backlog is accepted to be done accordingly to the DoD an Increment is born. An increment, that is another Scrum artifact, is an item of the product backlog which is defined completed and that satisfy the quality criteria accepted for the project. The Developers at the beginning of the sprint move into the sprint backlog a set of cards whose sum of the history points do not overcome the forecasted number of history points for the sprint. Moreover, the DoD helps the developers in understating the actual number of story point completed in a sprint. The number of history points completed in a sprint represent the sprint velocity. The average of the velocities of the sprints represents the velocity of completion throughout the duration of the project.

Scrum Artifacts	
Product Backlog	Product features items to be implemented.
Sprint Backlog	Subset of the product backlog: Items to be implemented throughout the sprint.
Increment	Any item of the Product Backlog that is considered "Done" accordingly to the DoD.

Table 2 – Scrum Artifacts

4.3.2.2 Kanban framework

As for Scrum, Kanban is a framework for the Agile Project Management. Kanban is a best practice also utilized in Lean manufacturing based on the use of cards, which is the letteral translation of Kanban from Japanese. Those cards, in Lean manufacturing, are used to signal the consumption of goods providing thus a just in time replenishment policy.

This framework is not based on iterations with predefined task to be completed in a time-based sprint but on the concept of doing the tasks as soon as the previous one is completed, thus matching the team's capacity. Since it is not based on timed iterations, there is not a Sprint backlog but there is a single backlog which represent the Product backlog. The idea of maximizing the team's capacity and the absence of the sprints allows to react to the any changes as soon it arrives without waiting for the start of the following iteration to address the modifications.

Main components of Kanban:

- List of work: issues or task to be completed, also named stories.
- Columns or lanes: used on the Kanban board to distinguish task between different workstreams, users or projects.
- Work in progress limit: rule that limit the amount of work based on the team capacity.

In Figure 4.4 an example of Kanban board is reported, it is organized in columns and, if necessary, as above mentioned in lanes. The stories follow the lanes form the "Backlog" column up to the "Completed" one. The logic behind this board is really simple, one task can be moved from the "Backlog" to the "WIP" if the number of stories in the "WIP" is below the limit, the WIP threshold limits the rate at which the stories are completed. The WIP threshold is based on the team's capacity. Cards, as represented in the figure below, usually are identified with an ID which ease the management and their sorting to the different team members, cards can be assigned to one or more project team members.

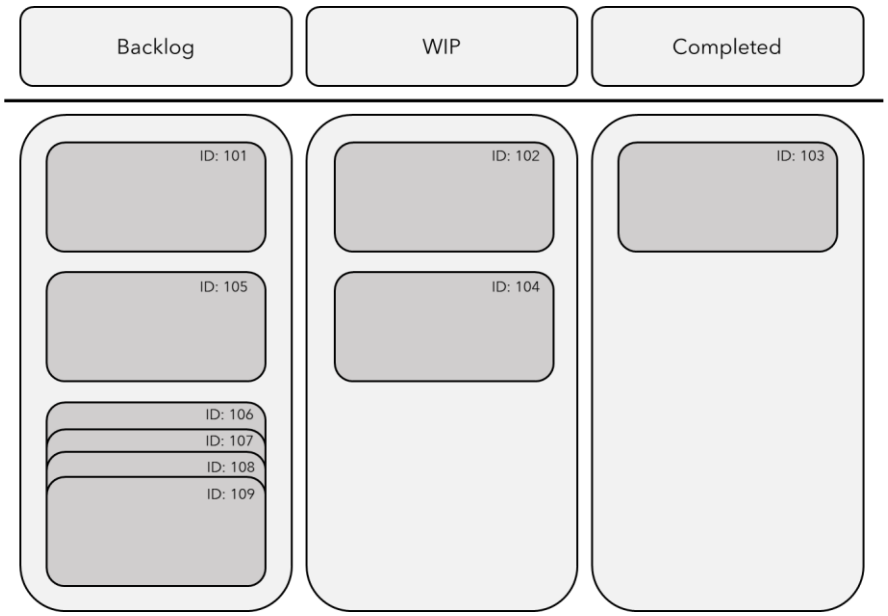

Figure 4.4 – Kanban board basic representation

4.3.3 Agile in PLM

"Product complexity and technology innovation may affect the use of APM practices for NPD environments" [21]. NPD complexity originates from several sources, for

example, technology uncertainty, number of components, systems and subsystems, and number of organizations involved, that undermine the project's team ability to deal with evolving requirements. "These challenges make the adoption of APM as a pure approach risky when compared with traditional NPD stage-gate (Waterfall) models, which assume a great deal of effort dedicated to the initial planning phase to identify and detail requirements and product specifications" [21][25]. For these reasons, to manage NPD the use of the APM in a PLM implementation project is unusual. The same level of complexity as for an NPD project is expected in the implementation of the PLM system, due also to the fact that the PLM system customization can be seen as a new product development itself. "The APM can generate rework, failures, and cost overruns in highly integrated products with interdependent components and systems" [21][26].

The software through which the PLM is implemented are not usually suitable for an Agile approach. For Early BOM project, as stated in Section 2.2, two software were used. ThingWorx Navigate, system of engagement, seems more suitable with respect to the APM approach thanks to its interface which allows multiple developers to work on it concurrently and allowing for, not always easy, modification of the GUI. On the other hand, for what concern Windchill, system of records, the APM is not suitable. Windchill represents the backbone of the system of applications developed in ThingWorx Navigate, it is used to manage all the information and data models on which each application is relying on.

To setup this system programming skills and knowledge are required. The modification of something on the system is not so straight forward. For example, the simple modification of the name of a soft-type attribute takes time and if there are already existing workflows or processes connected to that, everything should be checked again to ensure that still works. Moreover, while working on this software there are some best practices to follow, that are suggested by the software house, to ensure the best performances of the systems. These best practices involve also

ThingWorx Navigate, condition that cause the more APM friendly software of the two to do not suit the APM correctly anyway.

4.4 Adopted solution

4.4.1 Overview

In the Early BOM project the management approach used is a hybrid solution between the classic Waterfall approach and the Agile Scrum framework. In Section 4.3.3 the main reasons for which APM is unusual on PLM projects are listed; Agile implies the management of a lot of modifications throughout each sprint, changes that sometimes are no easy to be implemented due to the Early BOM subject. With this hybrid approach the idea is to reduce the number of non-value-added activities, thus being "leaner", guaranteeing, at the same time, the feasibility with respect to the project subject that usually is managed with the Waterfall model.
The project management approach is based on the time-based Scrum iterations framework, with integration of some hints coming from the classic Waterfall management approach and the Kanban APM framework.
In the Early BOM project, to avoid too many modifications on the system, prior the beginning of the first development sprint, differently for what is done in the APM classic Scrum framework, one or more sprints were dedicated completely to the analysis to assess the current "AS-IS" (how processes and business logics are carried out before the PLM implementation).
These analysis sessions can be seen as the requirement gathering phase of the classic Waterfall approach. In these sessions the company which is providing the implementation of the PLM system try to capture the current workflow and requirements. The aim of a PLM system is to simplify and improve the already existing processes but to do so there is the need to understand with a certain level of

detail the actual workflows and data-models. These meetings are held with the key users with the aim of understanding what are their expectations and requests.
Once the AS-IS analysis sprints end, the development sprints start. We should consider that, since for the PLM we cannot strictly use the APM, the sprint backlog is always open and cross sprints. Due to the necessity of respecting PTC best practices and the difficulties in making changes, the sprint backlog can recall activities done in previous sprints or anticipate activities for future ones. For what concern the sprint backlog, it is managed applying a mixed solution between Scrum and Kanban framework for APM. The details of the management of the backlog are reported in the next section with a more phase focused analysis.
At the end of each development sprint a meeting is held to align the Client key users on the functionalities under development and to be sure to focus on the one which are relevant to them; short demos on the developed functionalities are showed during these meetings. The demo is driven by the ACN project team and is not yet stable to be tested by the key users directly. This scenario is different from the classic APM approach, usually at the end of each development sprint there is the release to the key user and the testing of the increments. The handover of the new implemented functionalities is delayed to the end of all development sprints for a particular App, that is similar to what happens in the Waterfall model.
In the Early BOM project due to the hybrid project management solution adopted to overcome the PLM limitations, as above described, the release to the key users and the testing of the application functionalities is restricted to a dedicated project milestone, called Go Live. The Go Live, which is managed as a sprint, is fully dedicated to the release, testing and validation of the application developed.
During the Go Live, the Client have to validate what was developed ensuring that what was in scope of the project was delivered. To perform the validation and testing of the applications, ACN is providing to the Client a document which contains the

structure of the gathered use cases, during the first project phase, and the set of tests to perform the UAT.

An important remark to be noticed is that, differently from what happens in pure APM projects, there is plan for the release of the functionalities during the Go Live sprints. Usually for what concern APM the number of sprints is based on the product backlog that means that till the minimum number of features is reached, sprints are iterated, there is not a deadline that state the exact date for the release of the minimum features agreed with the clients. In the Early BOM project there is a plan that stresses the rhythm with which each App should be released. This approach limits the APM flexibility. If more development sprints are required, to satisfy the requirements, meetings between the leads of ACN and the Client are held to agree upon a plan change. The presence of a plan is a characteristic that is usually more relevant in Waterfall projects.

4.4.2 Project management phases

4.4.2.1 Analysis sprint

The first phase of development for each App is dedicated to the AS-IS analysis. These analyses should be considered as sprints where there is no developing, but the Product Owner and its key users are interviewed, by the analysis team, on how they are currently working. In the APM there are not dedicated sprints to the analysis sessions, this approach is similar to the Waterfall Requirements gathering phase. The team which participates to the sessions is made of senior developers/solutions architects which try to gather the main functionalities and start thinking for a possible user interaction with the system. Client key users are asked to participate to the meeting since they own the knowledge about the company workflows and business processes. During the analysis sprint usually about three meetings were held,

the first one was primarily used to define which are the Client workflows, starting from the second one proposals of the UI are showed to the key user. The mockups of the UI are high level representations of the users' interaction with the system. It was noticeable that the discussion arose on the mockups was useful to individuate those hidden use cases and workflows that at the first meeting were forgotten or not mentioned.

At the end of the analysis sprints a final recap meeting is held to review the identified use-cases, if agreed the development starts.

During the development of the Early BOM project, one or more additional analysis sprints were added where needed. This situation occurred for some of the applications developed in the project whenever the Scrum team felt that the level of understanding of the workflows was not sufficient to start the development.

4.4.2.2 Development sprint

The development sprints are managed following the APM approach. The used framework is the Scrum. Following the Scrum team structure described in Section 4.3.2.1, the following figures are identified:

- The Product Owner is represented by the stakeholder of the Client IT department.
- The Developers role is represented by the ACN development team.
- The figure of the Scrum Master is not clearly identifiable. In the management of the project a solution architect/developer manager was found providing the Scrum Master's duties internally for the Scrum team.

The absence of the figure of the Scrum Master is a criticality. There is the need to allow people that are working on the project to focus on their tasks without worrying about the project management, and foremost to lead correctly a project with the idea of applying APM practices the Scrum Master is fundamental to ensure the correct

adoption of the Agile Scrum framework. Accordingly to the applied Agile framework, sprints are on a temporal basis and at the beginning of each sprint the developers team declares to the Product Owner and the key users, in a meeting, what is the backlog for the sprint and thus the activities that are going to be performed at each sprint.

The items of the sprint backlog are managed accordingly to the selected Agile framework in a Scrum Board. The items in this board are used during the restitution meeting to show to the Product Owner and to the Key Users which were the activities/product features that were completed in the sprint and thus those new features shown in the demo during the sprint closing meeting.

For what concern the Developers activities, those are treated separately in a Kanban board. In this second board, Developers, starting from the activities summarized in the higher-level Scrum board, fill the Kanban backlog with those activities and tasks to implement the solutions both front-end (ThingWorx Navigate) that back-end (Windchill).

The presence of two boards based on two different Agile frameworks is abnormal. The management of the stories was adjusted to cope with the PLM environment by differentiating the Agile frameworks on which the two boards are based on. It is preferred to treat the Developers activities with a Kanban board since it allows to easily add items to the backlog, whilst as we have seen in Section 4.3.2.1 if there is the need to add a story while using Scrum this step is restricted to the Backlog Refinement ceremony.

The opening of a new card in the sprint backlog is uncommon in Scrum, usually the activities are moved from the product backlog to the sprint backlog at the beginning of the sprint, but with PLM there is the need to implement new activities to be sure to respect the software suite best practices and the systems integration. The higher level of flexibility achieved with the Kanban framework allows to counteract any

problem or any issue raised by the Client by opening a new card, thus managing a sprint backlog which is always open and cross-sprint.
This solution presents a drawback, if to implement one of the activities moved into the sprint backlog in the first board there is the need to open an unexpected number of cards in the Kanban Developers board, the amount of activities to be completed by the Developers to satisfy the Scrum sprint backlog might overcome Kanban WIP threshold, thus creating a bottleneck and slowing down the remaining activities. The team to avoid this problem tried to implement the history points logic seen in Section 4.3.2.1 for the Scrum framework in the Kanban framework thus replacing the WIP threshold logic typical of the Kanban. The use of the story points in the Kanban framework tries to sync the development board managed with the Kanban framework with the time-based iterations that stress the rhythm of the overall project.
For what concern the adopted mixed framework, the number of sprints is forecasted on the number of activities in the product backlog trying to respect the project plan to reach the Go Live in time. If the amount of work is higher than the one forecasted or due to delays one or more sprints are added to adjust the development time with the amount of work. At the beginning and at the end of each sprint, meeting with the key users and the development team are held to align all the members and to collect feedbacks.

4.4.2.3 Go Live

The Go Live represents the milestone at which the developed App is delivered to the key user to start the UAT and the start of the first adoption by the key users. ACN at the Go Live milestone delivers to the key users the developed solution and a document called “Skeleton” that collects the use cases identified in the applications and the related tests to verify that they are working as expected with respect to the company workflows and systems. The developed solution for the UAT is deployed in

the quality environment, that is a copy of the IT environment in which later the new developed functionalities will be adopted.
The tests identified represent all the possible interaction of the users with the system and thus obviously the consequences of these actions. The document delivered to the Client contains test scripts that the key users are going to perform to validate the delivered solution, each script describes step by step the interaction of the users with the system.
In the classic APM approach this phase does not exist, usually at the end of each development sprint there is the testing and the deployment of the new deliverable functionality in the client IT systems. In the Early BOM project the approach is similar to what happens in the Waterfall approach in the Testing phase.

4.4.2.4 Adoption

It is the last phase, the application, accepted by the key users, is released to all the users and adopted at company level. Concurrently to the rolling out of the application, the training material is released too. The training material validity is discussed in the third phase, during the UAT. The adoption of the developed solution starts different months later with respect to the Go Live milestone. The solution accepted and tested by the key users during the Go Live might be adopted before the official adoption by the testers themselves that allows them to report any issue and problems that might be related to the long-term implementation with the other company systems.

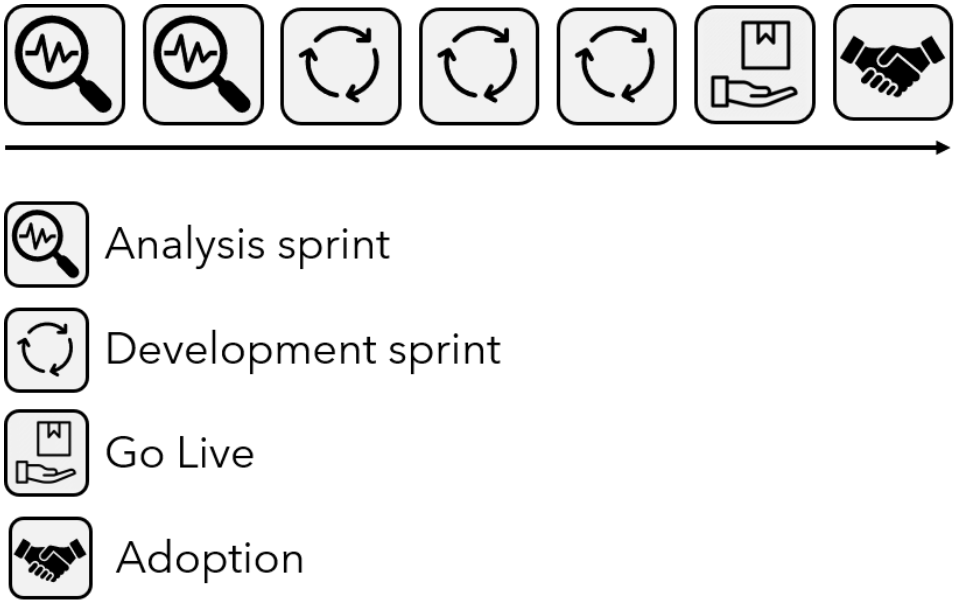

Figure 4.5 – Approach scheme

In Figure 4.5 a scheme of the phases is reported. At the beginning one or more sprints are dedicated entirely to the analysis sessions; once completed the analysis the forecasted number of development sprints start till reaching the Go Live, milestone which signs the first handover to the Client. Lastly the developed solution once validated and accepted is adopted.

The flow just described is applied individually to each application to be developed in ThingWorx Navigate. Each application/subject has its own dedicated stream of sprints, allowing parallel asynchronous development, that translates in the chance to perform analysis sprints on one subject whilst the development team is already working on another application development sprint. Figure 4.6 is a fictional example of the organization of the sprints for the overall project.

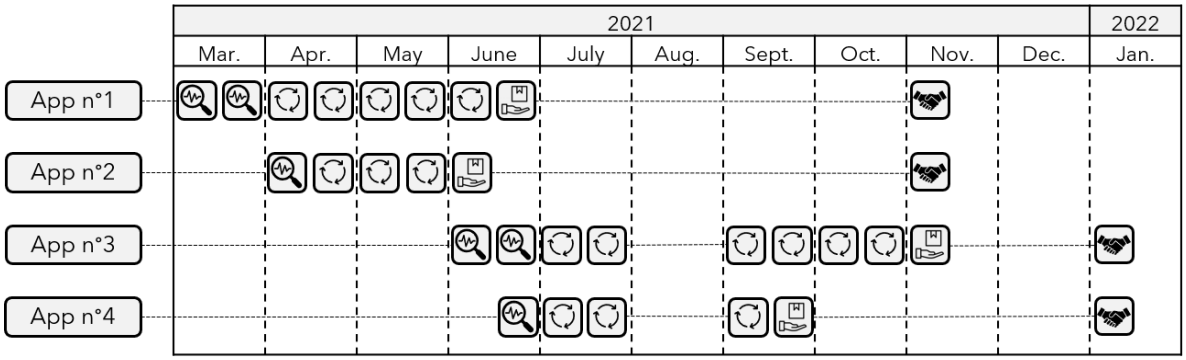

Figure 4.6 – Project management Gantt representation

4.4.3 Project solution and Lean thinking

In this section a brief analysis on the project management approach with respect to Lean thinking is reported. Referring to Mary Poppendieck, "Principles of Lean Thinking" paper, the analysis focuses on the identification on how much the applied project management model sets back with respect to the basic principles of Lean software development. At this point of the paper, it should be clear to the reader that APM model is not suitable for PLM projects. APM by its nature is more complaint to the Lean thinking concept, the implementation of some of the APM methodologies and tools with the Waterfall model allow to tackle a PLM project in a more "leaner" way.

In Mary Poppendieck paper, she translates the concepts of Lean, that usually are applied to the manufacturing processes, to the software development. The implementation of a PLM system does not exactly represent a software development project; to deliver the solution the project team is relying on already existing PTC software, but due to high level of customization that is achieved by generating customer specific UI on ThingWorx Navigate the project can be analyzed, and thus considered, as a software development project.

The Basic Principles of Lean Development
Add nothing but value (Eliminate Waste)
Center on the people who add value
Flow value from the Demand (Delay commitment)
Optimize across organizations

Table 3 – Principles of Lean Development [10]

4.4.3.1 Add nothing but value

The first principle of Lean development recalls the concept of waste introduced in Section 4.1. Waste is everything that do not add value to the product or service that is under development, it is crucial to understand how to assess what is value and what in not. If something is considered a waste it should not be present. Value is anything that is perceived by the customer as a product feature or service related to the product, for example equipping a city car with expensive compound tires makes no sense, the driver will never drive that car to max out the performances of those tires. This feature is a waste, both in term of money and time spent in the development for a compound dedicated to that car.

As described in Section 4.1 the term Muda identifies seven types of waste: Over Production, Waiting, Transportation, Over Processing, Inventory, Motion and Defects. In Mary Poppendieck, "Principles of Lean Thinking" paper each category is translated to the software development.

The Seven Wastes of Software Development		
Over Production	=	Extra features
Waiting	=	Waiting, inclunding customers
Transportation	=	Handoffs
Over Processing	=	Extra steps
Inventory	=	Requirements
Motion	=	Finding Information
Defects	=	Defects not caught by tests

Table 4 – Software development seven wastes [10]

In Table 5, a resume on how the Early BOM model behaves with respect to the identifies wastes is reported. The second column of the table labeled “Model” refers, for each one of the seven wastes, to the approach used in the Early BOM project for the specific waste.

	Model	How Early BOM adresses waste
Extra features	APM	The Scrum Team moves from the product backlog to the sprint backlog the activities of the sprint. There is no way that a developer performs an activity that introduces extra features with respect the ones agreed with the Product Owner.
Wainting, inclunding customers	APM/Waterfall	The waiting waste in not removed. In Early BOM it is not possible to deliver with small increments the developed functionalities. At the end of each development sprint the project team shows a demo of the developed functionalities, which is a partial delivery to the customer.
Handoffs	Waterfall	The handoff of the applications is bounded to the Go Live milestone, that represents the first contact point of the key users with the developed system. The transportation waste would have been removed if the delivery of the developed solutions was performed continuously.
Extra steps	APM	Over processing might arise from those development activities, added during the development sprint, required to respect the PLM software logics and best practices. New cards are added to the sprint backlog, cards that might cause the overprocessing waste.
Requirements	Waterfall	The analysis sprints performed at the beginning of each development, gather, accordingly to the waterfall model, an "inventory" of requirements to be satisfied. The waste is not removed.
Finding Information	APM	At the end of each development sprint meetings are held to gather feedbacks and issues raised by users. The motion waste is removed, it is easy to gather the information required.
Defects not caught by tests	Waterfall	The defects waste is not removed. Due to the project model applied it is not possible to test the developed app with the users until the Go Live with the UAT. Tests are performed by the project team during the development sprints to test the stability of the system but the system can be considered completed and accepted only after the key users UAT.

Table 5 – Early BOM waste management

For what concern the Early BOM project management approach accordingly to the resumed information in Table 5, the management of waste is far from being accomplished. The only wastes that result removed from the analysis are: Extra Features and Finding Information, that translated back to the Lean vocabulary are respectively Over Production and Motion.

4.4.3.2 Center on the people who add value

Taiichi Ohno, the mind behind the Toyota production system, understood that to add value to the product and to remove wastes the role of the employees was crucial. In the Toyota plants he gradually assigned to the production workers managerial/engineering tasks. Assigning tasks with a certain level of responsibility to the line workers, push them to fell more participant and enhance their skills. The idea is to train these workers in doing these engineering tasks to improve employee problem solving and mistake proofing. The higher the level of independence of the line workers the larger the time saved on the engineers to complete low skilled task. Accordingly to Mary Poppendieck, the same principle can be applied in software development. If the low skilled programmers are independent and able to follow a structured process in the development, the easier is for higher level managers and architects designing the solution.
For what concern the Early BOM project this approach is found. During the first phase of analysis, the team that participates to the meetings is made of solution architects and senior developers. The knowledge of these figures allow them to design a mockup UI which represents the solution to be shown to the key users to assess if the use cases and requirements are going to be satisfied. Meanwhile during these sessions, the remaining of the development team can keep working on the system. The development team and the analysis team are strictly in contact throughout the

entire development, supporting each other to enhance the performance of the entire project team.

4.4.3.3 Flow value from demand

For what concert Lean, one of the main pillars is the idea of having a flow of value which is pulled from the demand whenever needed, Kanban methodology is based on this logic which allows to develop a JIT replenishment policy.
"In Lean Software Development, the idea is to maximize the flow of information and delivered value. As in lean production, maximizing flow does not mean automation. Instead, it means limiting what has to be transferred, and transferring that as few times as possible over the shortest distance with the widest communication bandwidth as late as is possible" [10]. The concept is the one of reducing the number of information/delivered value handoffs limiting those to the right moment, right place and the right cost. To achieve this goal "the single most effective mechanism is delivering increments of real business value in short time boxes" [10].
In the standard Scrum APM framework the development is performed on temporal iterations that allow to manage the flow of information accordingly to the JIT approach, the information for the development of a sprint are gathered at each sprint. For what concerns Early BOM project due to its particularities, the assessment of the information is performed mainly at the beginning during the analysis sprints. As stated in Section 4.4.2.1 the first phase of AS-IS analysis is similar to the Waterfall approach, it is thus possible to state that the Early BOM model does not apply to the JIT concept for what concern the information flow. On the other hand, the number of documents generated is smaller with respect to the number of documents generated during the development of any other project applying the standard Waterfall approach requirement gathering phase since other data and feedbacks can be collected easily in between sprints, in particular during the end sprint reviews.

4.4.3.4 Optimize across organizations

One of the biggest problems trying to achieve a good value flow is that many times companies are organized with a heterogeneous structure. For example, metrics to measure the performance from one department to the following one are not the same. The misalignment of the departments cause turbulence in the flow of value, one department might optimize its performances damaging the overall company performance. Lean organizations, to solve this problem, aims at reducing as much as possible the handover of the business value form one department and the other; to do so usually the project team is made of people coming from different departments. The approach is known as the autonomous team structure; these teams are made of members coming from different departments, they refer to a team leader which usually is a senior manager [27]. This structure gives to the team a high level of independence, due to their freedom in managing internally the passages from one department to the other; they manage their own processes and working practices. The same principle translates to the software development, "development teams are best structured around delivering increments of business value, with all the necessary skills on the same team" [10]. In the Early BOM project the development team is managed accordingly to the above-mentioned practices. The team is made up of ten people, how suggested in the Scrum guide: "The Scrum Team is small enough to remain nimble and large enough to complete significant work within a Sprint, typically 10 or fewer people" [23]. The Early BOM team is cross-functional, meaning that the members have all the skills necessary to complete the project tasks.

5 User Acceptance Testing analysis

5.1 UAT

UAT stands for User Acceptance Testing, it represents a common process of any IS development project. This process is carried out at the end of the development with the aim of understanding if the business values has been delivered correctly when operated by its end users through tests. The UAT focuses on the acceptance of the overall system, for what concerns Early BOM project due to the subject this is a core process to be carried out. The implementation of the PLM solutions is attained by developing an infrastructure based on API on different business logic on two different software, which requires a detailed testing phase to ensure the correct functioning and integration of the whole system.

The UAT is performed usually in a quality environment. The quality environment is a replica of the adoption/production environment for the developed product. The use of this environment is preferred, in such a way the risk level of releasing something which is not working perfectly and that does not integrate sufficiently well is mitigated. The quality environment allows to the client UAT team to perform the set of tests without worrying about the damages that they could create if something goes wrong. The tests that the users perform during the UAT phase is the closest they get to the actual developed solution before the official adoption.

"What we need for UAT is a formal kind of testing that will enable us to check that a system does everything it is supposed to do and nothing that it is not supposed to do" [28]. The testing procedure must be as much as possible formalized, the stricter is its structure the easier is to drive the key users through the tests and the gathering of their issues.

The following three bullet points [28] report the definition of the main steps of the test development process.

- Test condition: an item or event of a component or system that could be verified by one or more test cases, for example a function, transaction feature, quality attribute, or structural element.
- Test case: a set of input values, execution preconditions, expected results and executions post conditions, developed for a particular objective or test condition, such as to exercise a particular program path or to verify compliance with a specific requirement.
- Test procedure specification: a document specifying a sequence of actions for the execution of a test. Also known as test script or manual test script.

For what concern Early BOM project the three above bullet points can be translated as reported in the following table.

Test Development Process	
Test condition	AS IS use cases applied by means of the developed UI
Test case	User interface driven test cases
Test script	Skeleton document

Table 6 – Test development process

The test conditions are represented by the AS IS use cases, gathered during the analysis sprint, that were transposed to the developed UI. The use cases explained by the users represent the project business value, that with the UAT the project team wants to ensure to have correctly delivered.

The test cases are strongly user interface driven. The users interact with the system only by means of the developed front-end ThingWorx Navigate applications. The test case identified are based on data entry and interactions via the screen.

The test script is represented by a document called “Skeleton”, it represents the backbone of the UAT, it contains the steps to perform the tests but also it is structured such to maintain the reference to the test condition and to ease the gathering of the test result. To help the client UAT team this document contains the reference to the training material too. Accordingly to the UAT logic, the users which perform the tests have been already introduced to the developed solution and to them the training material is just a tool to integrate the test scripts. Furthermore, the gained experience of the key users during the UAT and the developed documentation are the basis for the generation of the training program that the Client deploys at the adoption of the system.

5.2 Skeleton

This section is dedicated to the description of the document that the project team provided to the UAT team to perform the testing activities. The file was developed by the joint collaboration between the Product Owner and the project team leader. It is called “Skeleton” since it represents the structure of the testing activities, with this Excel file tests results are gathered too.
The first sheet of the file is a table which resumes the test cases, a brief description of the use case and some information about the role of the users are reported. The last information is a link which redirect the users to the test script sheet detail. As stated in the previous section, the test case number is the reference to find in a shared folder the documentation developed as training material.

#	Topic	Use Case Overview	Rif. Test	Role
1	**App N°1**			
1.1	Creation of …	Creation of a document from the main page of the UI…	T11	R&D
1.2	Edit of …	Check out the document…	T11	ALL
1.3	Research of …	Apply filters and…	T12	Marketing
2	**App N°2**			
2.1	Creation of …		T21	
2.2	Edit of …		T22	
2.3	Research of …		T22	
3	**App N°3**			
3.1	Creation of …		T31	
3.2	Edit of …		T32	
3.3	Research of …		T32	
4	**App N°4**			
4.1	Creation of …		T41	
4.2	Edit of …		T42	
4.3	Research of …		T43	

Figure 5.1 – Skeleton structure

More than one test case if needed refer to the same test script. A test script can be developed to test more than one test case at a time, due to the fact that some test cases are preparatory/mandatory to test the other ones. For example, in Figure 5.1 the test case 1.1 and 1.2 both refer to the test script T11.

In the test script detail sheet, the UAT users find the steps to take to test a functionality. Each set of actions refer to one of the test cases identified in the first sheet, mind that more than one test case can be ascribed to the same script detail sheet. Beside each step there is a column that reports the expected results for each action, the information reported in these cells helps the user in understanding if the procedure has been correctly performed and thus in identifying the possible errors.

T11	1.1 Creation of… 1.2 Edit of…

				Test User 1		Test User …	
TEST CASE	**ACTOR**	**ACTIONS**	**EXPECTED RESULT**	**Test result**	**KO description**	**Test result**	**KO description**
1.1 Creation of…	**R&D**	1. Access the link: https://www.polito.it/	Your are redirect to the Polito home page	OK		OK	
		2. Click on login.					
		3. Login with your credentials					
		4. …					
		5. Create a new …	The just created item is the folder named …				
1.2 Edit of…	**ALL**	6. Click on the top left of the page the edit button		KO	The edit button does not respond.	OK	
		7. Modify…					
		8. Write in the text field…					
		9. …					
		10. Confirm the changes and save…	A confirmation pop-up opens				

Figure 5.2 – Test script and result retrieval sheet

The users report the result of the test with a simple "OK" if the test is successful or with a "KO" if the functionality described in the test script does not work. Each user has its own dedicated cells, to allow to gather for the set of actions more than one test result. If the test result is a "KO" the UAT user must insert in the dedicated cell the error encountered. The results gathered during the Go Live UAT testing, are analyzed in the next section. The results of the tests are an assessment method to evaluate how good the project management approach applied in the Early BOM project is.

The issues found by the testers are numbered and treated as depicted in Figure 5.3. The tester raises the defect that later is assigned to the development team that validate the defect encountered. During the validation the development team check what is the defect and categorize it:

- if the defect is a bug or a software problem on an agreed requirement the issue is assigned to one or more developers which work on the code to solve the problem;
- if the defect reported is identified as a missing requirement the issue is categorized a change request (CR) and discussed with the lead of the project;
- defects that have been already reported by other users are categorized as duplicated;

Once the validated defect is fixed the developer hands to the architecture team the deployment of the fixed solution in the quality environment to be tested again. Firstly, the test is taken by the development team and subsequently by the same user which raised in the first instance the defect. If the test ends positively the issue is closed, if the defect is still present, the issue is reiterated accordingly to the workflow just described. The defects categorized as CR are discussed by the project team lead and the Product Owner. CRs are rejected if an agreement is not reached between the two parties involved. On the other hand a defect identified as CR, it is closed and the details of the approved CR are discussed on another table to specify

the changes in the contract between the company project team and the Client, specifying the increased amount of time required to implement the change and consequently the price of the changes discussed.

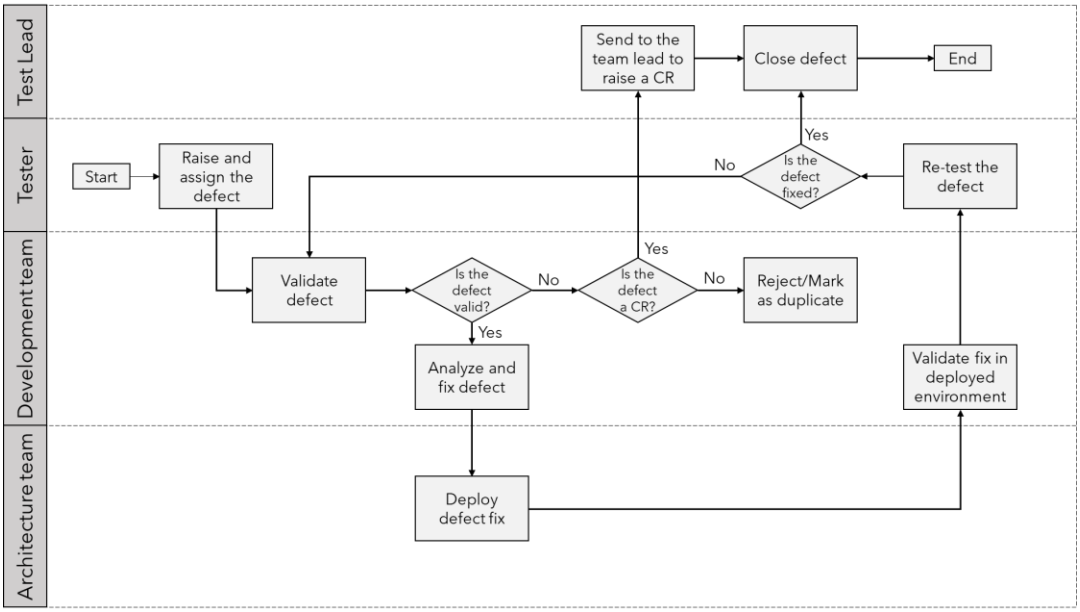

Figure 5.3 – Issues workflow

5.3 Result analysis

This last section is devoted to analysis of the results and to the description of the adopted statistical distribution to study the goodness of the applied project management approach.

5.3.1 Binomial distribution

The Binomial distribution is the discrete probability distribution of the number of successes in a sequence of n independent experiment of a stochastic variable that has

only two incompatible outcomes: success with probability p, and failure with probability 1- p. These experiments are known as Bernoulli's experiments.

Binomial distribution properties:

1. n is the prefixed number of Bernoulli's experiments.
2. Each observation can be categorized as success or failure, the two options are incompatible.
3. The probability of success, p, is constant for each observation, as it is constant the probability of failure, 1- p.
4. The result of an observation is independent from the result of any other observation.

$$P(x) = \frac{n!}{x!\,(n-x)!} p^x (1-p)^{n-x} \quad 5.1$$

Where:

- n is the number of independent experiments
- p is the probability of success
- 1- p is the probability of failure
- x is the number of successes

Thus:

- $P(x)$ is the probability of obtaining x successes with n independent experiments and p probability of success.

For what concerns the Early BOM UAT results, due to the only two possible outcomes of the tests (OK or KO) the Binomial distribution was selected. These results can be considered as discrete independent events. The observation of a failure on one test case do not influence the outcome of the next test. Thanks to the structure of the "Skeleton" document the sequence of the tests is such to stop the tester if one test case, that is preparatory/mandatory for the next one, fails. As described in

Section 5.2 each test script details the tests for more than one test case, we should mind that the results of the tests are gathered on each single test case and not for each test script. The tester performs the following test case only once the previous one has been solved, if failed at the first try; it is possible then to assume that the results of the test cases are independent.

The Binomial distribution applies to the UAT test results as follows:

- p is 0.5 since the possible outcomes are only two.
- 1- p then is equal to 0.5.
- x is the number of successes registered for a test case.
- n is the number of UAT testers that execute the test, more precisely the number of tests performed for a test case.

5.3.2 Chi square test

The χ^2 test compares how good the statistic distribution selected fits the actual observed results distribution. To test if the discrepancies, between the actual outcomes frequencies with the expect ones, are so small that can be attributed to the random variability (or not) we compute the χ^2 as follows:

$$\chi^2 = \sum \frac{(f-e)^2}{e} \qquad 5.2$$

Where:

- f the observed frequencies
- e the expected frequencies

The attained value of χ^2 is then compared to the value of the χ^2 distribution coming from the table for a selected significance level and degrees of freedom. The significance level, denoted with α, is the probability to reject the Null Hypothesis

given that it is true. The Null Hypothesis (H_0) states that there is no real difference between the two compared populations. The number of degrees of freedom (dof) is computed as the number of terms of the χ^2 minus the number of quantities required to compute the expected frequencies. For what concern the UAT under study the required quantity to compute the expected frequency is only 1 thus the number of dof is the number of terms of Equation 5.2 minus 1 (if the number of terms is 4 since we have 4 distribution classes, then degrees of freedom is 3).

If the χ^2 computed is larger than the one obtained from the table, it is possible to state that the Null Hypothesis is rejected. On the other hand, if the computed value is smaller than the one coming from the table then it is not possible to reject the Null Hypothesis.

If we reject H_0 it is possible to state that the selected distribution (the Binomial one for the Early BOM project) does not fit the data correctly. If we accept H_0 the selected distribution fits the data.

Applying the χ^2 square test, the idea is to prove that it is possible to reject the Null Hypothesis and thus to state that the Binomial distribution does not fit the results. This would prove that the results of the UAT are not stochastics variables and thus they are not due to chance. Therefore, if it is possible to reject the Null Hypothesis, it is possible to state that the number of successes obtained is not random and that was achieved thanks to a good requirement gathering and delivering.

dof	0.995	0.99	0.975	0.95	0.90	0.10	0.05	0.025	0.01	0.005
1	0.000	0.000	0.001	0.004	0.016	2.706	3.841	5.024	6.635	7.879
2	0.010	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210	10.597
3	0.072	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345	12.838
4	0.207	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277	14.860
5	0.412	0.554	0.831	1.145	1.610	9.236	11.070	12.833	15.086	16.750
6	0.676	0.872	1.237	1.635	2.204	10.645	12.592	14.449	16.812	18.548
7	0.989	1.239	1.690	2.167	2.833	12.017	14.067	16.013	18.475	20.278
8	1.344	1.646	2.180	2.733	3.490	13.362	15.507	17.535	20.090	21.955
9	1.735	2.088	2.700	3.325	4.168	14.684	16.919	19.023	21.666	23.589
10	2.156	2.558	3.247	3.940	4.865	15.987	18.307	20.483	23.209	25.188
11	2.603	3.053	3.816	4.575	5.578	17.275	19.675	21.920	24.725	26.757
12	3.074	3.571	4.404	5.226	6.304	18.549	21.026	23.337	26.217	28.300
13	3.565	4.107	5.009	5.892	7.042	19.812	22.362	24.736	27.688	29.819
14	4.075	4.660	5.629	6.571	7.790	21.064	23.685	26.119	29.141	31.319
15	4.601	5.229	6.262	7.261	8.547	22.307	24.996	27.488	30.578	32.801
16	5.142	5.812	6.908	7.962	9.312	23.542	26.296	28.845	32.000	34.267
17	5.697	6.408	7.564	8.672	10.085	24.769	27.587	30.191	33.409	35.718
18	6.265	7.015	8.231	9.390	10.865	25.989	28.869	31.526	34.805	37.156
19	6.844	7.633	8.907	10.117	11.651	27.204	30.144	32.852	36.191	38.582
20	7.434	8.260	9.591	10.851	12.443	28.412	31.410	34.170	37.566	39.997
21	8.034	8.897	10.283	11.591	13.240	29.615	32.671	35.479	38.932	41.401
22	8.643	9.542	10.982	12.338	14.041	30.813	33.924	36.781	40.289	42.796
23	9.260	10.196	11.689	13.091	14.848	32.007	35.172	38.076	41.638	44.181
24	9.886	10.856	12.401	13.848	15.659	33.196	36.415	39.364	42.980	45.559
25	10.520	11.524	13.120	14.611	16.473	34.382	37.652	40.646	44.314	46.928
26	11.160	12.198	13.844	15.379	17.292	35.563	38.885	41.923	45.642	48.290
27	11.808	12.879	14.573	16.151	18.114	36.741	40.113	43.195	46.963	49.645
28	12.461	13.565	15.308	16.928	18.939	37.916	41.337	44.461	48.278	50.993
29	13.121	14.256	16.047	17.708	19.768	39.087	42.557	45.722	49.588	52.336
30	13.787	14.953	16.791	18.493	20.599	40.256	43.773	46.979	50.892	53.672
40	20.707	22.164	24.433	26.509	29.051	51.805	55.758	59.342	63.691	66.766
50	27.991	29.707	32.357	34.764	37.689	63.167	67.505	71.420	76.154	79.490
60	35.534	37.485	40.482	43.188	46.459	74.397	79.082	83.298	88.379	91.952
70	43.275	45.442	48.758	51.739	55.329	85.527	90.531	95.023	100.425	104.215
80	51.172	53.540	57.153	60.391	64.278	96.578	101.879	106.629	112.329	116.321
90	59.196	61.754	65.647	69.126	73.291	107.565	113.145	118.136	124.116	128.299
100	67.328	70.065	74.222	77.929	82.358	118.498	124.342	129.561	135.807	140.169

Table 7 – Chi square distribution

5.3.3 Computation development

The results of the tests were gathered from the Go Live sprint of two of the developed applications for the Client. Out of the 132 tests performed 114 were registered as "OK" and only 18 were marked as "KO".

The test cases were not always executed by the entire set of key users that the Product Owner identified as testers. For this reason, in the computation of the χ^2 different distribution classes were identified: it is possible to individuate test cases

tested by 2, 3, 4 or 5 testers. The computations of the χ^2 are carried on then considering these 4 different classes.

	Observed frequency f			
Number of successes	5 Testers	4 Testers	3 Testers	2 Testers
0	0	0	0	0
1	0	2	0	2
2	0	1	5	3
3	0	2	10	
4	1	8		
5	4			

Table 8 – Observed frequencies for each class

The steps and parameters to compute the χ^2 are detailed for the computations for the tests performed by five UAT testers, for the other classes the procedure is similar.

5.3.3.1 5 Testers

5 Testers				
Number of successes	Observed frequency f	Expected probability	Expected frequency e	$\frac{(f-e)^2}{e}$
0	0	0.031	0.156	0.156
1	0	0.156	0.781	0.781
2	0	0.313	1.563	1.563
3	0	0.313	1.563	1.563
4	1	0.156	0.781	0.061
5	4	0.031	0.156	94.556
	Σ 5			χ^2 98.680

Table 9 – 5 Tester χ^2 computation

In Table 9 the first two columns report the observed frequencies for the test cases done by 5 testers. The observed frequencies are reported for the six possible outcomes when considering 5 testers, from 0 to 5 successes.

The Expected probability is computed accordingly to the Binomial distribution as a function of x, number of successes with:

- p probability of success equal to 0.5.
- 1- p probability of failure equal to 0.5.
- n number of Bernoulli experiments equal to 5 as the number of testers.

The Expected frequency e is obtained by multiplying the Expected probability with the sum of the Observed frequencies, that is the overall number of occurrences. Finally, the last column computes for each row the terms that summed together give the value of the χ^2 (Equation 5.2). The computed value of the χ^2 for the considered number of testers is reported in Table 9.

From Table 7 is possible to gather the value of the χ^2 for a significance α equal to 0.05 and with a value for the degree of freedom equal to the number of terms of the χ^2 minus 1. Since there are 6 terms, as the possible success outcomes, and the only parameter that is needed to compute the Expected frequencies is the total number of occurrences. The number of dof is equal to 6 – 1 that is 5.

Computed χ^2	Table χ^2
98.680	11.070

Table 10 – 5 Testers χ^2

Since the computed χ^2 is larger than the χ^2 for a significance level of 0.05, the Null Hypothesis can be rejected and thus it is possible to state that the Binomial distribution does not suit the data.

5.3.3.2 4 Testers

4 Testers				
Number of successes	Observed frequency f	Expected probability	Expected frequency e	$\frac{(f-e)^2}{e}$
0	0	0.063	0.813	0.813
1	2	0.250	3.250	0.481
2	1	0.375	4.875	3.080
3	2	0.250	3.250	0.481
4	8	0.063	0.813	63.582
	Σ 13			χ^2 68.436

Table 11 – 4 Tester χ^2 computation

The Table χ^2 reported in Table 12 was gathered from Table 7 with α equal to 0.05 and dof equal to 4.

Computed χ^2	Table χ^2
68.436	9.488

Table 12 – 4 Testers χ^2

Since the computed χ^2 is larger than the χ^2, the Null Hypothesis can be rejected and thus it is possible to state that the Binomial distribution does not fit the data.

5.3.3.3 3 Testers

3 Testers				
Number of successes	Observed frequency f	Expected probability	Expected frequency e	$\frac{(f-e)^2}{e}$
0	0	0.125	1.875	1.875
1	0	0.375	5.625	5.625
2	5	0.375	5.625	0.069
3	10	0.125	1.875	35.208
	Σ 15			χ^2 42.778

Table 13 – 3 Tester χ^2 computation

The Table χ^2 reported in Table 14 was gathered from Table 7 with α equal to 0.05 and dof equal to 3.

Computed χ^2	Table χ^2
42.778	7.815

Table 14 – 3 Testers χ^2

Since the computed χ^2 is larger than the χ^2, the Null Hypothesis can be rejected and thus it is possible to state that the Binomial distribution does not fit the data.

5.3.3.4 2 Testers

2 Testers				
Number of successes	Observed frequency f	Expected probability	Expected frequency e	$\frac{(f-e)^2}{e}$
0	0	0.250	1.250	1.250
1	2	0.500	2.500	0.100
2	3	0.250	1.250	2.450
	Σ 5			χ^2 3.800

Table 15 – 2 Tester χ^2 computation

The Table χ^2 reported in Table 16 was gathered from Table 7 with α equal to 0.05 and dof equal to 2.

Computed χ^2	Table χ^2
3.800	5.991

Table 16 – 2 Testers χ^2

Since the computed χ^2 is smaller than the χ^2, the Null hypothesis is not rejected and thus it is possible to assume that the binomial distribution fits the data.

5.3.4 Outcomes

Number of testers	Computed χ^2	Table χ^2
2	3.800	5.991
3	42.778	7.815
4	68.436	9.488
5	98.680	11.070

Table 17 – χ^2 Outcomes results

Table 17 resumes the values of χ^2 above computed for the 4 distribution classes identified. It is possible to state that the Null Hypothesis can be rejected for those tests that were done by 3, 4 and 5 users. On the other hand, for what concern the test developed by only 2 testers it is not possible to reject the Null Hypothesis.

The Binomial distribution fits the test cases performed by two tester and thus the data can be described accordingly to the Binomial distribution data characteristics: discrete probability distribution of the number of successes in a sequence of n independent experiment of a *stochastic variable* that has only two incompatible outcomes: success with probability, p, and failure with probability 1- p.

The χ^2 test was performed to prove that the data do not fit the Binomial distribution, so to state that the data are not stochastically distributed and that the applied management approach is capable of gathering and satisfy the Client requirements. Since the Null Hypothesis cannot be rejected for the test cases performed by only two testers it is not possible to state that the data are not randomly distributed.

On the other hand, for those test cases, the majority, performed by 3, 4 and 5 testers the Null Hypothesis can be rejected and thus it is possible to state that the data are not stochastically distributed.

The fact that the Null Hypothesis cannot be rejected for the tests performed by only two testers is mainly due to the low number of total observed frequencies and the fact that almost half of these few observations falls in the middle class which accordingly to the Binomial distribution, with n equal to 2 and x equal to 1, has an expected probability of 50%; that is exactly what to be expected if the data are independent stochastic variables with only two possible outcomes.

It is interesting to note that also the tests performed by 5 users have the same number of total observed frequencies of the tests performed by only two testers, but thanks to the higher number of success classes and the distribution of the observations towards the high end of these success classes the Null Hypothesis can be rejected accordingly to the χ^2 test.

Conclusions

Throughout this paper the Early BOM management approach was detailed highlighting its peculiarities and criticalities. The bibliography research performed on the standard management approaches, on which the Early BOM approach is based, points out how the applied solution does not respect what is established as the management approaches state of art. The adopted approach is a hybrid solution between the Waterfall and Agile Scrum model. Although the applied model does not respect all the rules and methodologies of the literature models, from the analysis turned out that the solution was capable of satisfying and gathering all the Client requirements for the development of the PLM applications.

First, it is important to recall the subject of the project Early BOM, the implementation of an integrated PLM system to foster the generation of a generic BOM that allows to sync and keep updated different systems and departments within the Client organization. The Hybrid solutions was adopted to cope and counteract the challenges linked to the PLM implementation.

Indeed, if the project would have been a "standard" software development project, the deployment of an APM framework and strategy would have worked straight away, exploiting all the framework potentialities and tools. In Early BOM the only adoption of APM would not have been satisfactory, leading to the management of a project where there would have been the need of recalling already closed backlog items to be reworked or tuned to collaborate with other successively developed applications features. Closed backlog items, accordingly to the APM framework selected (i.e. Scrum or Kanban), are released as soon as possible or at the end of the sprint, if later there would have been the need to recall already closed items the Client would have felt a continuous interruption of the developed functionalities that would have slowed down the business operations and processes without bringing any

added value. Furthermore, the continuous and several service interruptions, required to proceed with the development would have been felt by the Client as a lack in the management capability by ACN.

On the other hand, the adoption of some of the Waterfall management phases, as for example the requirements gathering phase, were fundamental to address those project items that should be addressed with a blueprint, since of primary importance, to ensure the success of the overall project. Due to the complexity of the PLM system, there is the need to deepen from the beginning some aspects regarding the IT infrastructures and business logics that connect the different developed applications, thus ensuring the correct development of the project goals. It is obvious, as described in Section 4.4.3, that the hybrid solution does not respect Lean best practices due to the mixture of tools and methodologies coming from the less "Lean" Waterfall management model. Even though the applied management model is far from being Lean, the team took advantage of those Lean practices and features that from the analysis turned out to be respected (see Table 5).

Section 5 focuses on the analysis of the results of the UAT. The results of the UAT were used to assess the goodness of the management approach in satisfying the Client requirements. The UAT results are considered as a metric to evaluate the outcome of the project. The outcome of the analysis of the tests is good enough to state that the project management approach, together with the team skills, were capable of driving and managing the project phases with a positive result. Data and test results were collected from two Go Live sprints, within the issues registered as "KO" tests cases none of those were related to core functionalities; no main requirements were left back or left completely unconsidered. The issues reported by the UAT testers are mostly related to UI details or fine tuning (i.e. the position of a button in the UI or the sequences of clicks to achieve a determined action). Other issues were related to the accesses, within the developed applications different roles, with different privileges, were identified accordingly to the Client department organization. Those

accesses issues were promptly solved since there was the need for the user to first log to ThingWorx Navigate to be successively profiled in the quality environment where the tests were performed.

In conclusion, the Hybrid management approach, despite its criticalities, turned out to be a good solution to face Early BOM project. ACN took advantage of the two literature models to create an approach that suits the project subject the best. First, the use of time-based iterations kept in touch the Client Product Owner with the Team, this practice fostered the generation of a working relationship based on the collaboration to achieve the best possible result. Moreover, the closer the Client is to the consulting company the easier it is to gather the information and requirements. Lastly, the adoption of a sprint (Go Live) entirely dedicated to the UAT ensures that the developed functionalities and features are working and behaving correctly with the Client IT infrastructure.

References

[1]. Bernadette Law, Professor Stephen Evans, Understanding Luxury in the Premium Automotive Industry, Building 50, School of Applied Sciences, Cranfield University, Cranfield, Bedford, England, MK43 0AL.

[2]. Caronaro, S.: The Pyramid of Luxury and "Democratization" of Luxury (abstract). De-signing Designers Conference (2006)

[3]. McKinsey&Company, The road to 2020 and beyond: What's driving the global automotive industry?

[4]. https://www.faistgroup.com/news/automotive-sector-change-post-covid-19/ available in 2021-04-26

[5]. Valentina Gecevska, Zoran Anisic, Paolo Chiabert, "Solution of collaborative product lifecycle management" ACTA TECHINCA CORVINIENSIS – Bulletin of Engineering, Tome IV (year2011)

[6]. Bernard A., Tichkiewitch S., Design of Sustainable Product Life Cycles, Springer-Verlag, 2008 - X. G. Ming, J. Q. Yan, Technology Solutions for Collaborative Product Lifecycle Management - Status Review and Future Trend, Journal Concurrent Engineering, Vol.13, No.4, 2005, pp.311-321.

[7]. X. G. Ming, J. Q. Yan, Technology Solutions for Collaborative Product Lifecycle Management - Status Review and Future Trend, Journal Concurrent Engineering, Vol.13, No.4, 2005, pp.311-321.

[8]. Joseph Ngae, "Accenture PTC Ecosystem Overview", KX documents available in 2021-04-14

[9]. H.M.H. Hegge and J.C. Wortmann, Generic bill-of-material: a new product model

[10]. Mary Poppendieck, Principles of Lean Thinking, http://sel.unsl.edu.ar/ApuntesMaes/Anteriores/MetodologiasAgiles/LeanThinking.pdf available in 28/09/2021

[11]. Peter Himes, Matthias Holweg, Nick Rich, Learning to evolve: a review of contemporary lean thinking

[12]. https://theleanway.net/muda-mura-muri available in 2021-06-01

[13]. Anna Corinna Cagliano, Plants manufacturing systems B lecture notes, Basics of Lean Manufacturing

[14]. https://www.umsl.edu/~hugheyd/is6840/waterfall.html available in 2021-05-29

[15]. http://www-scf.usc.edu/~csci201/lectures/Lecture11/royce1970.pdf available in 2021-05-29

[16]. https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm available in 2021-05-29

[17]. Oriana Karina Eason, Information Systems Development Methodologies Transitions: An Analysis of Waterfall to Agile Methodology

[18]. https://www.giuneco.tech/uat-e-ui-testing/ available in 2021-05-30

[19]. History: The Agile Manifesto (https://agilemanifesto.org/history.html) available in 2021-07-13

[20]. Manifesto for Agile Software Development (https://agilemanifesto.org/) available in 2021-07-05

[21]. Edivandro C. Conforto, Daniel C. Amaral, Agile project management and stage-gate model – A hybrid framework for technology-based companies

[22]. Itisha Singh, White paper on AgileProjectManagement, KX documents available in 2021-04-14

[23]. https://scrumguides.org/index.html available in 2021-07-10

[24]. Mensch, Caroline, Accenture Interactive Virtual agile Workshop, KX documents available in 2021-04-14

[25]. Vinekar, V., Slinkman, C.W., Nerur, S., Can agile and traditional systems development approaches coexist? An ambidextrous view.

[26]. Turk D, France R, Rumpe B, Limitations of agile software processes.

[27]. Peerasit Patanakul, Jiyao Chen, Gary S. Lynn, Autonomous Teams and New Product Development

[28]. Brian Hambling, Pauline Van Goethem, User Acceptance Testing - A step-by-step guide