# SCRUM++ Framework concepts

Régis Plateaux, Olivia Penas, Sagar Mule, Peter Hehenberger, Stanislao Patalano, Ferdinando Vitolo

# SCRUM++ Framework concepts

PLATEAUX Régis
*Quartz Laboratory, EA7393*
*SUPMECA*
Saint Ouen, France
regis.plateaux@supmeca.fr

PENAS Olivia
*Quartz Laboratory, EA7393*
*SUPMECA*
Saint Ouen, France
olivia.penas@supmeca.fr

MULE Sagar
*University of Applied Sciences Upeer Austria*
Wels, Austria
sagar.mule@fh-wels.at

HEHENBERGER Peter
*University of Applied Sciences Upeer Austria*
Wels, Austria
peter.hehenberger@fh-wels.at

PATALANO Stanislao
*Fraunhofer J-Lab IDEAS*
*University of Naples Federico II*
Naples, Italy
patalano@unina.it

VITOLO Ferdinando
*Fraunhofer J-Lab IDEAS*
*University of Naples Federico II*
Naples, Italy
Ferdinando.vitolo@unina.it

***Abstract*— Traditional design processes must adapt to new industrial challenges, to the rapid evolution of technologies and the resulting complexity of systems. Today's industry, particularly in the field of mechatronics, must design and develop ever more innovative products while reducing time-to-market in order to maintain a competitive edge. As late changes during the realization and detailed design phases lead to a considerable increase in costs and design time, it is necessary to introduce more flexibility during the development process. The agile approach has already proven successful in the design of software system and offer many benefits, as it aims to limit the rigidity of the specifications, interfaces and organization, and to involve in a more flexible way the different actors, customers, specifiers and partners. In this context, we propose a MBSE approach to identify the set of requirements related both to the mechatronic product development and to the dynamic market, companies and current new trends, in order to define the SCRUM++ framework key concepts that aim to meet previous requirements, by supporting agile hybridization methods.**

***Keywords—Agile Hybrid Approach, Mechatronic Product Development, Agility, SCRUM.***

## I. Introduction to Agile Methods

In these turbulent times, when most industries, customers' expectations and technologies are rapidly changing, it is not only needed to adapt to changes but also to accept them, to consider them as an opportunity to improve products, industries and the use of latest technologies, while satisfying customers' needs [1]. These changes, digitization, uncertainties in product development, customers as dominant stakeholders and their inability to specify all the requirements upfront, have given rise to the agile methods. The agile manifesto was defined in early 2001 in order to align all developed agile methods [2] [3]. The most widespread agile method is Scrum.

The term Scrum originated for software development in 1986 by Takeuchi and Nonaka by referring to scrummaging in a rugby game. Indeed, similar to a rugby match, where the whole team has to work towards a common vision and goal, while offering flexibility according to the conditions of the game, agility has become a vital characteristic in a development process [4]. Therefore, Scrum is an interesting method that can be applied not only to software but also to hardware product development [5]. Indeed, many products surrounding us nowadays do not come from a single domain or a mere combination of software and hardware but from the complex integration of multiple domains. Even though software is currently dominating products, hardware is still a support of the software and adds fundamental value to the product [6]. Therefore, the application of agile methods for the mechatronic industry is starting to appear as an interesting research topic. Scrum can be used for Mechatronic products, but some aspects still demand adaptions, e.g. for critical system development, the combination of different domains, the organization of the corresponding multidisciplinary team and the resulting complexity [1].

This paper aims at describing the Scrum++ framework concepts, as an efficient way to introduce agility into the mechatronic product development.

### A. Motivation for Agile Design

The fast pace in development of technologies, innovation, sudden changes in market, increasing complexity in the architectures and concepts of products has awaken the fear of failure. To address these fears, the term VUCA was presented first in 1987. In fact, it is rather a picture of opportunities to structure organizations in such a way that will meet the challenges presented by the environment than a fear, the terms of which must be understood [7]. VUCA stands for Volatility, Uncertainty, Complexity and Ambiguity. The VUCA framework has been explained in Table 1. Most uncertainties are related to the inability to predict the changes in the market, and though agile methods cannot predict the future trends, it can adapt any changes during the development process. This aspect helps organizations to be ahead in adapting any changes in requirements and technology.

*Table 1: VUCA framework adapted from [7]*

| | What is it | Example | How to address it |
|---|---|---|---|
| **Volatility** | Unpredictability of change | Cost of jet fuel is volatile nowadays | With agility |
| **Uncertainty** | Lack of significant knowledge of an event and its ramification | Anti-terrorism program is filled with uncertainty, no idea about when and how it will happen. | Gaining information, analyzing information |
| **Complexity** | Complex nature of information | Moving to another market, complex rules and regulations | Restructuring company to be ready for complexity from outside world |
| **Ambiguity** | No idea about the expectation | From print to digital, companies are coping with how customer will experience this. | Experimentation needed, customer involvement |

### B. Scrum

Scrum is an iterative method that focuses on the development of a set of tasks in each sprint (predefined short cycle), influenced by product vision as a common goal, iterative learning and self-organization [8]. Among the three internal roles, the product owner is responsible for the "what" in relation with the stakeholders, the Scrum master for "how" to development the product, and the development team [9]. The three pillars of Scrum are transparency, examination and adaption. These pillars make Scrum effective by allowing information sharing on daily activities and development progress, frequent inspections to know issues about the product at an early stage and faster reactions on the feedback from the inspection in order to develop a suitable product [10]. Fig. 1 provides a brief overview of artifacts, activities and roles within Scrum [11], [12].

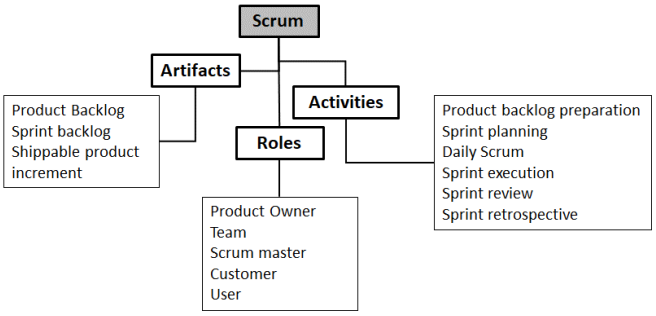


Fig. 1. Roles, artifacts and activities in Scrum

### C. Current research on Scrum-integration for mechatronic product development

After the introduction of Scrum for software development, many organizations and researchers have proposed some changes to make it suitable for mechatronic products development. Some of these recent research works are detailed here.

According to Böhmer et al. [11], for mechatronic product development, implementation of Scrum within the METUS diamond (linking the elements of the functional structure to those of the product structure), together with engineering change management (ECM) provides benefits such as the prioritization of functions and the modularization before the development, identification of critical system elements, early delivery of the core product, etc. This method helps reduce uncertainties in the product development by providing picture of the product architecture before development while maintaining the important aspect of agility which is adaption of any changes during development in order to develop an innovative product [11].

Mabrouk et al. [4], conducted research in the area of mechatronic product development using Scrum with Model-Based System Engineering (MBSE). In order to define the product backlog (set of system requirements), they use a MBSE black box analysis to provide an exhaustive set of requirement that limit risks related to future changes while offering the flexibility to adapt any requirement change at the same time. This ability reduces unnecessary changes and associated cost by a complete requirements analysis beforehand. This framework has been built to reduce the difficulties in mechatronic product development, while using the nominal Scrum method, which makes it easy to adopt and use [4].

Other few modification of the product development method with the use of Scrum are, Scrumban [13] and the Scrum-Stage-Gate method [14].

### D. Agile methods challenges and limites for Mechatronics

The SCRUM methodology is not a clear set of rules and guidelines for development but a framework, which results in self-regulation.

Even if the strength of this framework is that it allows weaknesses of the product to be directly unmasked, it can result in friction during the development process. In addition to that, the multidisciplinarity of mechatronics complicates the application of SCRUM, as the assignment of team members is more difficult, since it is almost imperative to include all the relevant disciplines.

Another challenge is that typical mechatronic products often are subject to laws (e.g. emission laws) and have to fulfil norms and certification requirements. All of these require a significant amount of documentation, which is often neglected in SCRUM, because it is not explicitly demanded. Additionally, mechatronic companies usually distinguish between structural and functional organization. This possibly creates tensions between the Team, which is in a superior position regarding autonomy and responsibility and the rest of the company. In parallel, retaining high reliability of mechatronic product needs a well-designed critical system and Scrum framework does not have any function that can solve the issue of system criticality [1]. However, the use of tools such as design structure matrix (DSM) can support it. The issue is then how to integrate these tools before prioritizing the backlog and to analyze which measures have to be taken to develop the critical elements of the system. Also, to test these critical elements, prototyping is an important tool which can help customers understand the system well and faster [15]. A last limit would be, as in first few iterations of mechatronic product development, prototyping of the fully functional product is not possible, it might restrict the customers to provide a relevant feedback on the system quickly (even though the team has completed tasks as per the definition of the "done" (product delivered at the end of each sprint cycle)).

Existing literature shows that, due its specificities, mechatronic products development requires some agile methods adaptations to be more efficient. Depending on the context, the market, the companies, such adaptation can be achieved through many kinds of hybridization between traditional and agile methods.

## II. RESEARCH MOTIVATIONS

Our research work focuses on the development of a methodological framework aiming at integrating agility in the development of mechatronic products. As it seems complicated or worse impossible to integrate all agile aspects in mechatronic product development within the previously described context, we propose an object-oriented approach, which inspired the name of the proposed framework SCRUM ++, in reference to the "C++" language concepts. This framework will propose some concepts points of view that could be combined together at the desired moment.

The objective of this paper is first to identify all the requirements that such a framework has to both fulfill the mechatronic products development requirements and the constraints related to the dynamic market, companies and new

trends. We then describe the Scrum++concepts that agile hybridization methods have to include in order to meet the previous set of derived requirements and then to achieve successful agile mechatronic product development. Finally, we present the pedagogical case study that initiated this research interest and proved the value of hybridizing agile and traditional design methods.

## III. NEEDS ANALYSIS

Based on the black box analysis of the SE-READ (System Engineering based Requirements Elicitation & Architecture Design) approach [16], we have identified the derived requirements generated by the coupling of the mechatronic products development constraints and those of the current market trends (Fig. 2).

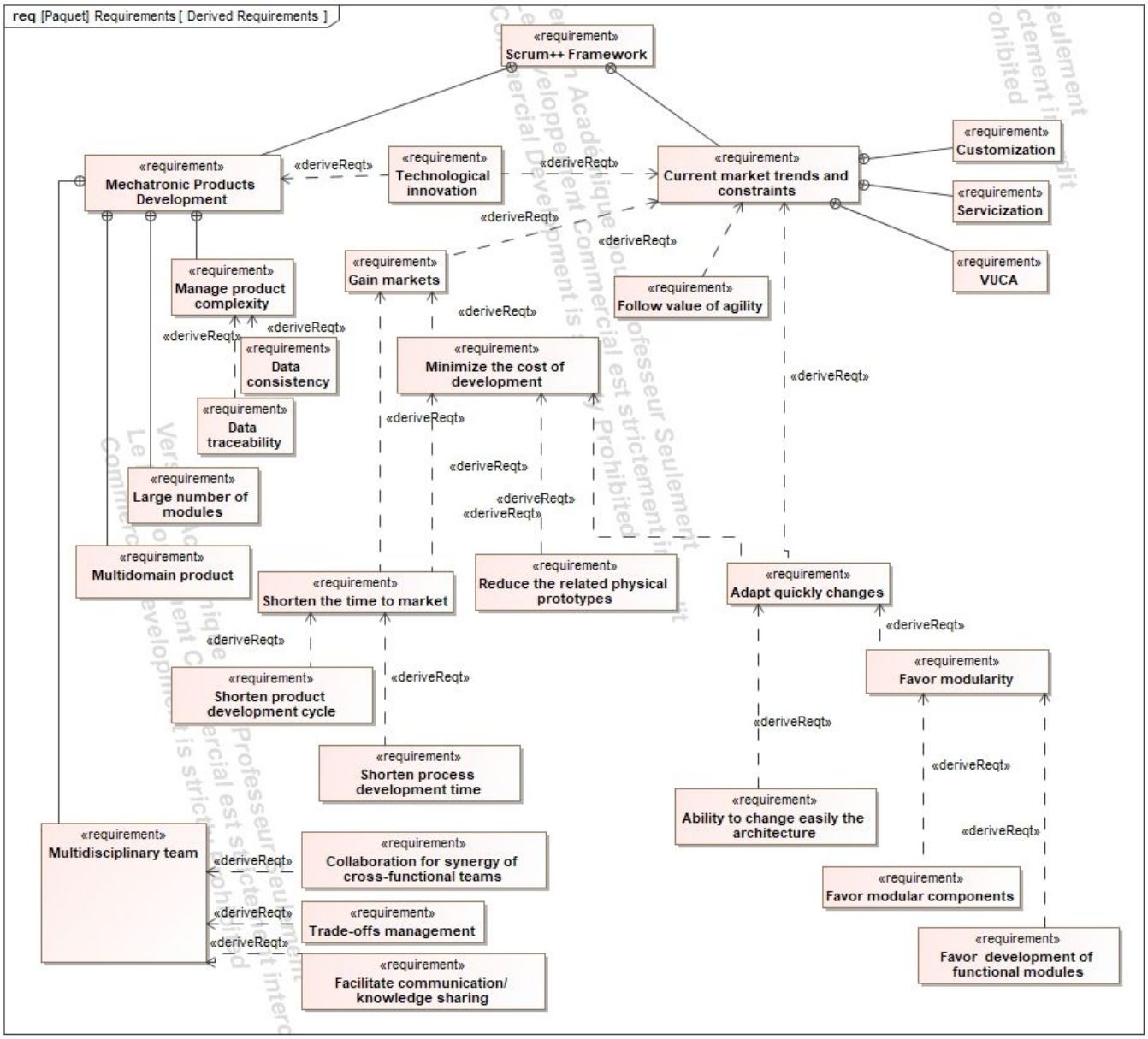


Fig. 2. Scrum++ framework requirements

## IV. CONCEPTS OF THE SCRUM++ AGILE HYBRIDIZATION FRAMEWORK

The objective of the framework we propose to develop is to meet the previous derived requirements. In other word, it aims to enhance the mechatronic products development in a market-driven context. Our main idea is to provide a framework, named SCRUM ++, that can provide agile hybrid approaches adapted to the needs and constraints of each company in the development of mechatronic products.

Based on a MBSE approach guided by the SE-READ method [16], we will describe how the various concepts of the SCRUM++ framework (Fig. 3) meet the previous derived requirements.

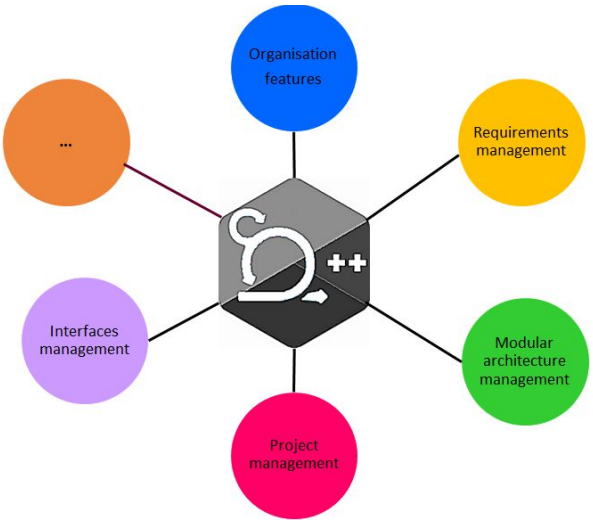


Fig. 3. Scrum++ current concepts

### A. *Organisation Features*

Before adopting a hybrid agile approach for the development of mechatronic (or multi-disciplinary) products that is relevant for organizations, it is necessary to clearly identify their needs and constraints. Indeed, the support of companies towards agility strongly depends on their structure, their expectations, the available resources, their development process, the mechatronic product to be developed, etc. Thus, the study of the changes management in the current product development processes and the associated organizational structure is a particularly interesting subject to explore when considering an agile hybridization of mechatronic products development. A part of this work was quickly presented in a previous paper [17] and the details will be provided in a future paper.

This study topic meets the SCRUM++ requirements detailed in Fig. 4.

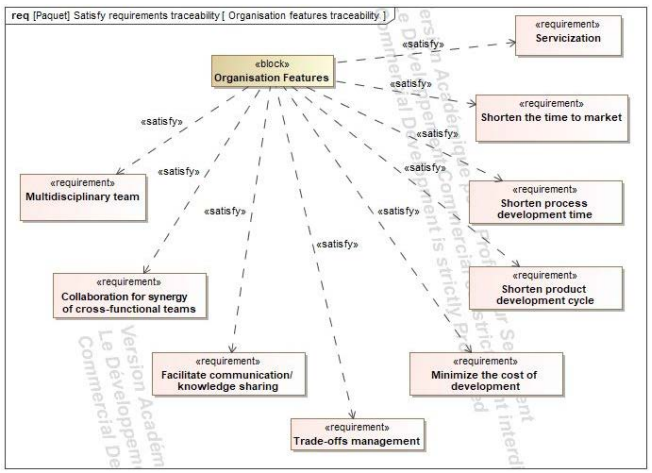


Fig. 4. Satisfy traceability provided by the Organisation features concept

### B. *Requirements Management*

In an agile approach, the requirements management presents the main following features: (i) the system requirements are defined all along the development process, (ii) they are prioritized by the stakeholders as the product development progresses and (iii) they are treated progressively in order of priority, with regular steps of verification and exchange with the stakeholders. Such properties allow to take into account the evolution of the customers' needs during the product development process.

In order to efficiently support requirements change during a hybrid agile development, we propose different approaches.

The first one, based on the Scrum method will support the iterative selection of the highest priority requirements for the customer in each new cycle to define the Sprintbacklog (list of tasks meeting a subset of requirements/needs that the development team must perform during a "Sprint", which lasts from one to four weeks). Based on the Stakeholder Value Networks [18] and the MIAM (Multidisciplinarity Integration in Agile MBSE) method [4], we will define an agile SysML profile that can select the requirements to be dealt with in the next sprint, based on the requirements priority defined in accordance with the weight determined by the involved stakeholders.

The second one relates to the use of graph theory to identify requirements dependencies and to cluster dependent requirements to be handled as much as possible together (in the same design cycle), and if not, to trace them in case of a change in requirements to check if the dependency is still justified [19], [20].

In parallel, we propose another approach to deal with the requirements change tracking, based on the CatSE (Category-based System Engineering) framework, and notably the MAFALDA (Methodology Analysis and Formalization bAsed on oLogs to Design Architecture) methodology. The MAFALDA is a formalization of the SE-READ approach [16] to ensure data consistency within middle-out and agile context. This approach based on the coupling of the category theory and ontologies is able to trace any derived requirement to its native stakeholders' need and is then an efficient tool for requirements change tracking. This work is detailed in a paper under submission.

The fourth one proposes to study the specification of the nature of the test case for the verification and validation of a requirement. The idea is to define criteria that will allow designers to choose whether a given requirement should be verified or validated by a virtual or physical prototype: for example, according to the degree of maturity of the product under development, the availability of physical components, etc.

This study topic meets the SCRUM++ requirements detailed in Fig. 5.

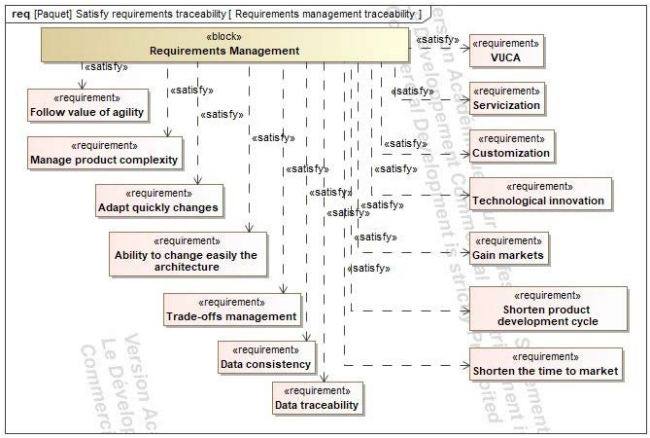


Fig. 5. Satisfy traceability provided by the Requirements management concept

### C. Modular Architecture

In parallel, as agile methods advocate short iterative cycles with the delivery, at the end of each cycle, of a (often partial) prototype (called "done" in the Scrum approach), it is important to study the modularity of the system throughout its development.

As product architecture consists of four elements: the functional architecture, the logical architecture and the physical architecture, and the associated allocations of elements from one to another, the system modularity can be addressed at these four levels.

We can propose to structure all the functions, as functional modules, based on an ontological approach, in order to define a functions library. The ontology benefits consist in overcoming the various terms used by designers, to relate only to the concepts associated with the corresponding functions. Finally, the functions modularity lies in the management of instances of the same function concept, by studying their properties (number of inputs/outputs, types ...) and using the semantic distance between the various terms. For this research work, we will also introduce the notion of "Function Identity Card" to define all the features characterizing a function.

The logical architecture includes the expected physical behavior of components, which can be materialized by some models in order to provide a virtual prototype of the logical architecture, as the final physical components are sometimes not yet available. At this level, the modularity will also be based on the ontologies related to the behavioral models. Similarly, we will use the existing "Model Identity Card" idea [21], to define the features identifying distinctly each model supporting the logical components. The definition of a modular virtual prototype composed of compatible logical modules and based on a simulation architecture build from a MIC architecture will provide an efficient support in agile approach to verify and validate the behavior of components regarding the set of requirements addressed during a development cycle, without having to choose a particular physical component at this stage (and thus keeping the modularity of the corresponding physical architecture).

The modularity of the physical architecture will be based on the COTS (Commercial Off-The-Shelf) management. Indeed, the cost reduction and the market customization trend lead companies to develop their product based on standard or reconfigurable components, in order to offer the largest products variants with the same COTS. The modularity at this level addresses the ability of the physical architecture to integrate different COTS, either by waiting for the optimal final component to be achieved as the development matures, or because there is a concern to maintain maximum flexibility with regard to possible future changes in requirements.

Finally, we can propose a matrix view to manage the respective allocation of components between two types of architecture. The matrix view, and notably the DSM and DMM approach could be very efficient to determine the modularity relating to the different allocation possibilities respecting the modularity of each of the architectures and interfaces of the different elements (clustering techniques).

This study topic meets the SCRUM++ requirements detailed in Fig. 6.

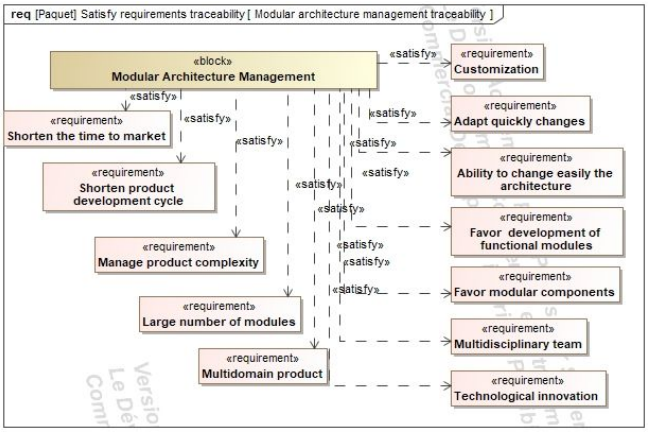


Fig. 6. Satisfy traceability provided by the Modular architecture concept

### D. Project Management

As original fully agile methods are hardly directly applicable to the development of mechatronic products for the reasons described in section I.D, we propose to merge the benefits of traditional and agile development methods.

Different hybrid design approaches have been identified, based on our own experience in an educational context, on experts interviews and a literature review.

Indeed, when comparing agile and traditional design methods, four main major differences appear. The first one address their related flexibility scope: whereas traditional design has a fixed goal and flexible cost and schedule, agile design is based on a flexible goal with fixed cost and schedule. The second one relates to the management of actors: in traditional design process, the project and decision management structure is hierarchical with an initial and final interactivity between stakeholders and designers, whereas agile methods offer a self-organization with a constant collaboration between them all along the development. The third difference relates that traditional approaches propose a linear development process with exhaustive requirements at start, whereas in agile methods the process is led by the iterative development of functional modules including the possibility of requirements change. Finally, whereas traditional project management focuses on the delivery of a complete and whole physical solution at the end of the development, agile methods can provide all along the design cycle some operational physical modules with limited but testable and upgradeable functionalities and a faster return on investment. These different aspects are all the more critical in the context of the mechatronic products development, because the collaboration and trade-offs between designers from different disciplines and domains also have to be managed, including an efficient teams' competencies management.

An example of such agile hybridization is given in a previous paper [17] and the case study below will illustrate the need for such hybridization.

This study topic meets the SCRUM++ requirements detailed in Fig. 7.

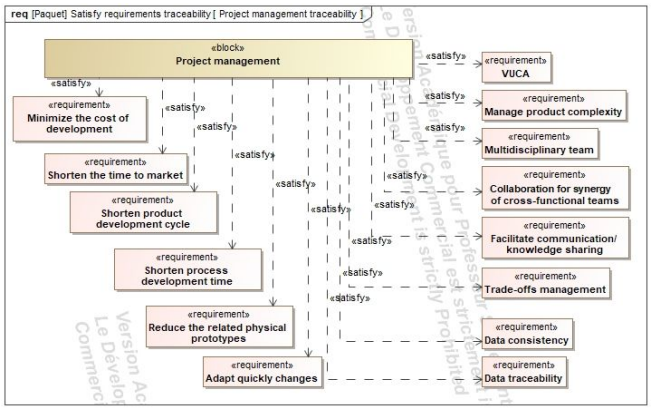


Fig. 7. Satisfy traceability provided by the Project management concept

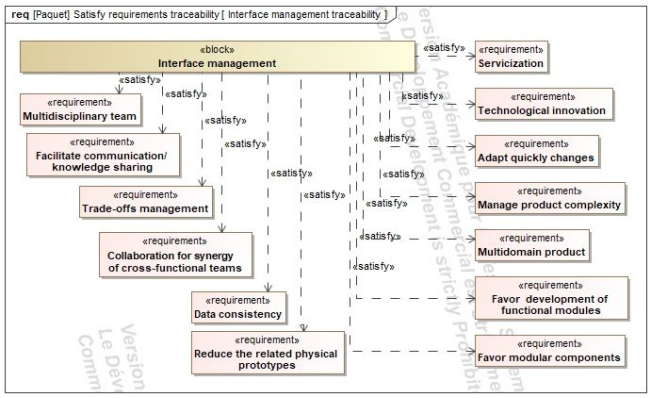


Fig. 8. Satisfy traceability provided by the Interfaces management concept

### *E. Interfaces Management*

As the project scope is often too large to be managed by a single agile team (ideally 7 to 9 persons), we have to study how to organize the interfaces management between several agile teams working on the same product to design. Preliminary reflections brought us to two areas for consideration: the Scrum of Scrum ($S^2$) notion and the definition of a new actor named Interface Manager.

Indeed, as it is very difficult, from our own experience, to perform a Scrum project with a high number of members (> 9 persons), we propose to divide the whole system development in different agile teams (with no more than 9 persons). The definition of the teams is made either by subsystems if the architecture concept has already been chosen, or by functions of the system to keep the maximum potential of modularity. Whatever the kind of teams breakdown, a certain number of measures, means and resources must be put in place beforehand, to efficiently manage both the information, data and knowledge share and the integration of the "done" from different teams all along the product development. As a result, we propose to define a Scrum of Scrum ($S^2$) framework including methodologies, tools, competencies management, etc. in a MBSE approach, and to describe the tasks of a key actor, namely the Interface manager, in charge to specifically manage all kinds of technical interfaces between the different teams within the whole project.

Finally, regarding the agile hybridization methods, the introduction, for each team's actor, of transversal roles coming from traditional methods, in complement with their agile roles, increase the information, data and knowledge consistency and traceability, in accordance with MBSE principles. This work will be detailed in a specific paper.

This study topic meets the SCRUM++ requirements detailed in Fig. 8.

### *F. Other aspects*

Among the other aspects of the Scrum++ framework that we have not yet begun to work on, there are still subjects that have questioned us and that will be worthy of being dealt with later on.

A first point relates to the definition of the Scrum++ process(es). Indeed, the question is: is there one or more methods to efficiently use the Scrum++ framework and depending on which criteria? The generated hybrid methodologies to manage the different phases of the product development (design-realization and testing) have also to be defined.

The second aspect corresponds to the level of abstraction on which one stands in agile hybridization: introduction of agile concepts in traditional methods or of traditional artefacts in agile methods? This position can have an impact on the way to use the Scrum++ framework.

The third additional research question focuses on the ability of the Scrum++ framework to be used in an analysis or a synthesis process.

Many other concepts could be and probably will be added to the current presented Scrum++ framework, to achieve a holistic view of an efficient mechatronic products development in a VUCA context. The oriented-approach inspiration should allow to integrate many other relevant research works.

Finally, to facilitate the acceptability of such a framework, the relationships between all the Scrum++ concepts and notably their interfacing need to be extensively studied, in order to identify the framework required features allowing to meet these concepts integration and interoperability issues. Sometimes, a third party (middleware) will be required for this integration approach. Additionally, some couplings could be identified and generate new requirements but also new benefits.

## V. EDUCATIONAL CASE STUDY

We have experienced the agile Scrum method during a project-based course named CSME (Conception et Réalisation des Systèmes Mécatroniques) for design and realization of mechatronic systems in English. It allows the students of the mechatronics option in Supméca-Paris in France to work together on a concrete subject in an agile context on a period of 48-96 face-to-face hours. This course has evolved over the years, based on feedback from teachers and students.

Initially, in 2017-2018, the objective of the CSME course was to develop a captive UAV for surveillance or inspection application, with an additional objective for the team to create a start-up. They therefore attended an entrepreneurship sensitization session led by an independent entrepreneur. A single team was composed of 9 students in a full SCRUM agile mode, the development project was based on an existing drone and they developed additional modular functionalities to detect forest fires. It was a successful mechatronic development, where stakeholders (teachers) and students were very satisfied.

In 2018-2019, there were 24 students who were divided into two teams of 12 students for the two sub-projects. The first one was to finalize the design of the previous year's project by developing a cable reel and a new feature: swimming surveillance with an onboard camera on the tethered UAV, and the second one concerned a modular robotic hand that could be attached to the tethered UAV to grasp objects. The two teams adopted the Scrum agile project management. But that year, the management of the project was a failure: there was no team cohesion, the teams did not interact with each other and developed their sub-system as two independent systems: the students were not interested by the entrepreneurial approach, they did not understand the interest of operating in agile mode, as they did not develop a team spirit, students regretted the lack of skills transfer among team members and the teachers were disappointed by the lack of overall vision and students interactions.

In 2019-2020, 15 students chosen to work on a new mechatronic product: a robotic modular platform able to participate in the French Robotics Cup, whatever the specific constraints and objectives of the current year's challenge. As the teachers wanted to keep the modularity and scalability of the system, each team of 7 students chose to focus on an essential and fixed function of the system: locating or moving. To address the weaknesses of the previous year's project, teachers introduced a new hybrid Scrum agile project management, based on four main features:

- The role of Scrum of Scrum manager has been assigned to the fifteenth student, in order to ensure the consistency of (i) the objective of each team at each sprint, (ii) the shared architecture/vision and the corresponding constraints, (iii) the global SysML product model.
- The introduction of an additional mission (for the global project related to more traditional design methods) to each team member (e.g. competencies manager, traceability manager, infrastructure resources manager…)
- The designation of two (one in each team) product interfaces manager, in charge to identify (i) all the interfaces of their sub-system to generate the corresponding requirements for the other team, (ii) the shared sizing parameters, in order to facilitate the trade-offs in case of non-compatibility of the respective sub-systems requirements.
- The competencies management has been introduced with an individual management of the skills (initial, desired and obtained) of the project members, which allowed a relevant allocation of backlogs to the students as the sprints progressed, in order to enable them to improve their skills and know-how in the areas they were interested in.

Finally, this last agile hybridization of the project led to more satisfaction from both students and teachers, demonstrating the interest of such agile hybridization for mechatronic product development.

## VI. Conclusions and Future Work

We propose the Scrum++ framework as a support to solve VUCA issues in the context of the mechatronic products development. Inspired from object-oriented approach, as it provides some building blocks to develop efficient agile hybridization methods, it is notably based on five primary concepts that we have begun to study: organization features, requirements management, modular architecture management, project management and interfaces management, but also on many other aspects that we have listed in this section IV.F, which will therefore constitute our future directions for work. This preliminary work will have to be tested on other case studies, in order to determine the quantitative results of such an approach. The development of tools to support the use and implementation of the Scrum++ framework will also be another challenge for future work.


## Acknowledgment

This work has been performed within the SUNISWELL consortium, supported by internal funding of each participating academic institution: University of Applied Sciences Upper Austria, Wels, Austria, Supméca, Laboratoire QUARTZ, EA7393, Paris, France and Fraunhofer J-Lab IDEAS, University of Naples Federico II, Naples, Italy.


## References


[1] D. Cohen, M. Lindvall, et P. Costa, « An Introduction to Agile Methods », *Adv. Comput.*, vol. 62, nº C, p. 1-66, 2004, doi: 10.1016/S0065-2458(03)62001-2.

[2] S. Denning, « The age of Agile », *Strategy Leadersh.*, vol. 45, nº 1, p. 3-10, 2017, doi: 10.1108/SL-12-2016-0086.

[3] N. Johnson, « Agile hardware development – nonsense or necessity ? », in *DESIGN West 2012*, 2012, p. 1-7.

[4] A. Mabrouk, O. Penas, R. Plateaux, M. Barkallah, J. Y. Choley, et A. Akrout, « Integration of agility in a mbse methodology for multidisciplinary systems design », in *4th IEEE International Symposium on Systems Engineering, ISSE 2018 - Proceedings*, 2018, p. 1-5, doi: 10.1109/SysEng.2018.8544425.

[5] R. G. Cooper et A. F. Sommer, « The Agile–Stage-Gate Hybrid Model: A Promising New Approach and a New Research Opportunity », *J. Prod. Dev. Manag. Assoc.*, vol. 33, nº 5, p. 513-526, 2016, doi: 10.1111/jpim.12314.

[6] C. Fuchs et F. J. Golenhofen, « Agile for Mechatronics and Hardware », in *Mastering Disruption and Innovation in Product Management*, Springer, Cham, 2019, p. 147-164.

[7] N. Bennett et G. J. Lemoine, « What a difference a word makes: Understanding threats to performance in a VUCA world », *Bus. Horiz.*, vol. 57, nº 3, p. 311-317, 2014, doi: 10.1016/j.bushor.2014.01.001.

[8] A. I. Böhmer, « When digital meets physical – Agile Innovation of Mechatronic Systems », Ph.D. dissertation, Faculty of Mechanical Engineering, Technische Universität München, Munich, Germany, 2018.

[9] R. Pichler, *Agile Product Management with Scrum*. Pearson Education, Inc., 2010.

[10] J. Cho, « Issues and Challenges of agile software development with Scrum », *Issues Inf. Syst.*, vol. 9, nº 2, p. 188-195, 2008.

[11] A. I. Böhmer *et al.*, « Towards Agile Development of Physical Products », *23th Int. ICE Conf. Eng. Technol. Innov.*, vol. 17, p. 78-85, 2017, doi: doi:978-1-5386-0774-9/17.

[12] J. Also, « Integration of the Scrum methodology in mechatronic product development », 2015.

[13] M. Stoica, B. Ghilic-Micu, M. Mircea, et C. Uscatu, « Analyzing Agile Development – from Waterfall Style to Scrumban », *Inform. Econ.*, vol. 20, nº 4, p. 5-14, 2016, doi: 10.12948/issn14531305/20.4.2016.01.

[14] R. G. Cooper, « Agile-stage-gate hybrids », *Res.-Technol. Manag.*, vol. 59, nº 1, p. 21-29, 2016, doi: 10.1080/08956308.2016.1117317.

[15] T. Gartzen, F. Brambring, et F. Basse, « Target-oriented Prototyping in Highly Iterative Product Development », in *3rd International Conference on Ramp-up Management*, 2016, vol. 51, p. 19-23, doi: 10.1016/j.procir.2016.05.095.

[16] F. Mhenni, J.-Y. Choley, O. Penas, R. Plateaux, et M. Hammadi, « A SysML-based methodology for mechatronic systems architectural design », *Adv. Eng. Inform.*, juin 2014, doi: 10.1016/j.aei.2014.03.006.

[17] S. Mule, R. Plateaux, P. Hehenberger, O. Penas, S. Patalano, et F. Vitolo, « A new agile hybridization approach and a set of related guidelines for mechatronic product development », présenté à IFIP 17th International Conference on Product Lifecycle, Rapperswil, Switzerland, juill. 2020.

[18] M. Sease, B. Smith, D. Selva, et J. Hummell, « Setting Priorities: Demonstrating Stakeholder Value Networks in SysML », in *INCOSE International Symposium*, 2018, vol. 28, p. 1207–1220.

[19] S. Patalano, F. Vitolo, S. Gerbino, et A. Lanzotti, « A graph-based method and a software tool for interactive tolerance specification », *Procedia CIRP*, vol. 75, p. 173–178, 2018.

[20] S. Patalano, F. Vitolo, et A. Lanzotti, « A digital pattern approach to 3D CAD modelling of automotive car door assembly by using directed graphs », in *Graph-Based Modelling in Engineering*, Springer, 2017, p. 175–185.

[21] G. Sirin, « Supporting multidisciplinary vehicle modeling : towards an ontology-based knowledge sharing in collaborative model based systems engineering environment », thesis, Châtenay-Malabry, Ecole centrale de Paris, 2015.