The Systems Engineering Approach as a Modelling Paradigm of the Agri-Food Supply-Chain

Maria Teresa Gaudio*, Sudip Chakraborty, Stefano Curcio

University of Calabria, Laboratory of Transport Phenomena and Biotechnology, Department of DIMES, Cubo-42a, 87036 Rende (CS), Italy]

mariateresa.gaudio@unical.it

The agri-food supply-chain represents a complex System-of-System (SoS), because it crosses different other sectors and involves many different actors. The System Engineering (SE) approach helps to identify boundaries as a line of demarcation between the system itself and its greater context, including the operating environment, without neglecting any aspect. Model-Based System Engineering (MBSE) was used, due to its capacity to realise more readable and compact documentation than other models. It uses System Modeling Language (SysML) to construct a structure, the behaviours, the requirements and the constraints of the system. The modelling method developed to design and implement a model is based on one of the main purposes of the agri-food SoS: the need of tracking and tracing useful information to realize a traceable and sustainable system. Agri-food supply-chain SoS was analysed through an iterative procedure model – developed in Papyrus open-source – that has the system requirements as the centre. The requirement diagram is composed of: consumer requirements, business requirements, legislation requirements and environmental requirements. The typical structure of the requirement package in SysML has inherent the parent-child relationship and responds in an appropriate manner to the system traceability. Papyrus can directly validate the designed model, without complex simulations, especially in the case of a high level of abstraction, which would be missing some implementation parameters necessary for the simulation of the technical phase. This type of validation allowed to check better and faster what has been developed through project simulations. This work presents the main results of the SE approach applied to the agri-food supply-chain. It represents a starting point for the choice of the technical traceability solution.

1. Introduction

Since ever, the technology was always a significant force in the agri-food supply-chain. From the first ox-drawn to the use of high-yield fertilizers, the inventions changed the agri-food, in particular the agricultural phase. But the agri-food sector does not end with the agricultural phase. It consists of different sectors that interact between them and it could be seen as a SoS, whose system elements are themselves systems; typically, SoS entail large-scale interdisciplinary problems with multiple, heterogeneous, distributed systems. Generally, SoS has the following characteristics: the operational independence of constituent systems, the managerial independence of constituent systems, a geographical distribution, emergent behaviour and evolutionary development processes. In other words, a SoS as the agri-food supply-chain exhibits complex behaviour: the interactions between the parts exhibit self-organization, where local interactions give rise to novel, nonlocal, emergent patterns. Thus, an holistic approach gave the possibility to always analyse the whole system, using an aspects approach, with an iterative procedure that can govern the complexity, which could also lead to an improvement of the whole agri-food supply-chain, in terms of sustainability. The boundaries of this agri-food supply-chain SoS are really difficult to identify if someone thinks with a classical systemic approach, because they may be confused with the subset of elements and subsystems that interact with the environment.

Traditionally modelling a SoS could lead to too extensive documentation, that would be very hard to realize and read. For this reason, the MBSE approach with others is actually in use for SE. MBSE improves the ability to capture, analyse, share and manage the information associated with different specifics of products, with the

following advantages: better communication between involved development parts; an increased ability to manage the system complexity, thanks to the system model visualization from a different point of views and as consequence, to analyse the impact of changes; a better product quality, thanks to a unified model that may be evaluated to different parameters; a better capture to knowledge and consequent reuse of information, thanks to the standard and incorporate abstraction mechanisms inherent to the model-driven approaches and thus, a reduced cycle time and reduced maintenance. Currently, a solution that ensures the traceability of the entire agri-food supply-chain does not exist and studying the system through a system engineering approach provides a possible generic global solution, able to model and update the information coming from the supply-chain. In the present work, MBSE was applied to analyse and model the entire agri-food supply-chain, in terms of traceability and consequent sustainability of the entire supply-chain. A modelling strategy was developed in SysML to construct an iterative procedure in Papyrus – an open-source edited by Eclipse – to ensure the traceability of the entire supply-chain. This procedure is mainly based on the requirements of the system, as the SE approach requires. Starting from an analysis of some specific agri-food supply-chains that characterize Italy and in particular, Calabria region, the main actors were identified and the forms were submitted to them for deriving needs and requirements from which to start to structure the resolution model. The requirements diagram represents the core of the SysML iterative model, which could update the system and give some indication about the technical solution. The model designed provides an immediate validation and check of it and at the same time, provides some indication about the possible technical solution. A more detailed level of abstraction could subsequently provide other values of performance, through the plug-ins of the tool.

2. Agri-Food supply-chain SoS Analysis

The analysis of the agri-food supply-chains to understand what are the similarities and critical points of the entire supply-chain, does not start from the point of view of internal or external logistics, but it refers to the concept of "integrated logistics", able to account for all phases and aspects of a production system to optimize it. Integrated logistics aims to combine the high quality and healthiness of the final product through the traceability of the process, with the highest possible environmental sustainability of the production process, to reduce or eliminate hazardous waste in processes and chemicals. The SE approach goes in this direction, recognizing and governing the interactions between different parts and actors of the system, avoiding neglecting some parameters and constraints.

2.1 Model-Based System Engineering

The MBSE was born in about the 1980s as a standard practice in electrical and mechanical design, when mechanical engineering transitioned from the drawing board to increasingly more sophisticated 2-D and 3-D computer-aided design tools. In parallel, the electrical engineering transitioned from manual circuit design to automated schematic capture and circuit analysis. Computer-aided software engineering became more popular in those years, using graphic models to represent software abstraction levels, above the specific programming languages. Consequently, the use of modelling for software development became more widely adopted, in particular with the advent of Unified Modeling Languages (UML) in the 1990s (Friedenthal et al., 2014). In subsequent years, the ever-increasing capabilities of technology at all levels have led to the adoption of the MBSE by more and more engineering disciplines. Thus, MBSE is considered a practice. The INCOSE System Engineering Vision 2020 of 2007, defined MBSE as "the formalized application modelling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases" (*Technical Operations International Council on Systems Engineering (INCOSE)*, 2007). To realize the practice of MBSE, there are three pillars: modelling languages, modelling methods and modelling tools. MBSE commonly uses the Systems Modeling Language (SysML) (Delligatti, 2014). The modelling method is a roadmap to plan the strategy to design and implement a model. Modelling tools are different. Most of these are commercial-grade and so, not free; only two of these are free modelling tools, offered with an Eclipse Public License (EPL) or with a General Public License (GPL): Modelio and Papyrus. Papyrus will be used for this work, because Modelio does not support SysML in the free version. In Papyrus, each model is created using a specific architectural context, even if it is possible to change it later, through the specification of the model details. A model also specifies a subset of the different points of view of its context and also each subset model may be changed later. As the first step, for creating a model there is a need to use a workspace with a specific project. The workspace is located in the file-system or a specific folder set by the user; when the tool starts, a pop-up window appears and the selection of the specific folder with the correspondent project may be opened.

The model created is the basis for all diagrams and at the same time, it maintains the consistency between the different diagrams, because it is the collection of different elements and their relationships defined in the different diagrams. In Papyrus, this collection is displayed in the Model Explorer view, where the different diagrams are

shown in their logical place within the model. This visual representation of a system that the different diagrams provide may offer both high-level or low-level insights into the conceptualization and design of a system.

To implement correctly a complete model, it is impossible to create a single diagram. For example, if a requirement diagram does not refer to a specific context, through the use of the other diagrams, it means that the requirements of whom or what would it satisfy? Therefore, it is necessary to implement other diagrams with a single diagram, to model a complete way a SoS, in each of its aspects. To model a SoS completely, at least three diagrams are needed: one between the behavior diagrams, the requirement diagram and one between the structure diagrams. Usually, for behavior diagrams, the use case diagram is used, because it represents the possible functionality that is performed by the system. A use case diagram is also able to identify the actors that interact with the system, but also with the external of the system, and also the relationships that exist between elements. The analysis of the agri-food supply-chain system goes into the direction of the life cycle management, starting from the technical and business point of view, because it is aimed at finding a possible technical solution to satisfy the System of Interest (SOI) – that is a SoS – regarding the aspect of traceability.

2.2 Stakeholders: needs and requirements

According to the ISO/IEC/IEEE 15288, the technical processes begin to develop based on needs and requirements of a system (Ryan, 2013). A stakeholder of a system is an individual or an organization with an interest in the system. A successful project depends on meeting the needs and requirements of the stakeholders through the system life cycle. Stakeholders are all those who may be influenced or influence the system, generally e.g., users, operators, regulatory bodies, and companies. The identification of the stakeholders carries the identification of needs with it. This process elicits the stakeholder needs that correspond to new capabilities or opportunities for the system. These needs are analysed and transformed into a set of the stakeholders' requirements for the working and the effects on the solution and its interaction with the working environments. The stakeholders' requirements represent the main reference compared to which an operating capability is validated. Thus, the requirements are measurable, unlike the needs. To obtain a good result, all critical points of the SoS must be taken into consideration, starting from the critical interactions between the different parts of the system. From various possible interactions of the system (Stoewer, 2003), the stakeholders' requirements govern the system development and they are an essential factor to define the scope of the development project. To elicit the stakeholders' needs, various techniques can be used e.g., interviews, focus groups, the Delphi method, and soft system methodology; also, the simulation tools may be used to evaluate different mission alternatives. These procedures could give back a partial view of needs and as a consequence, of requirements, if each aspect and interaction is not taken into account. The initialization of the requirements database and the development of the life cycle concepts serve to build a scenario useful as a methodology for planning and decision-making in a complex system and an uncertain environment. Each requirement carries with it a cost and so, it is essential to have a minimum set of requirements in the early project life cycle. According to the Standard ISO/IEC/IEEE 29148, the requirements definition is both iterative and recursive: iterative between the main process and others, when other information is available during the analysis; recursive because this process is continuously applied to define the requirements for each system element.

2.3 System Engineering Approach for an Agri-Food Traceability Model

Due to its complexity, the agri-food supply-chain may be analysed and treated through the system engineering approach. As shown in Figure 1a, through three steps, it was possible to lead a formulation of requirements of the entire supply-chain, to develop a model in SysML, where the traceability of information is the main objective. In this context, the analysis of some typical Calabrian agri-food supply-chain highlighted a market where the small and medium companies represent most of them and where these companies have similar sectors. Therefore, similar stakeholders internal to the companies – farmer, producer, packer and distributor – were identified among the final consumers. The variety and complexity of the SoS lead to the identification of only two major stakeholders: the companies and the final consumers. The second step consists of the elicit of the stakeholders' needs, with the following formalization of the stakeholders' requirements. This was made through two different questionnaires submitted to companies and final consumers. For companies, a statistically acceptable sample was not reached and thus, the information about companies was used as an indication in the requirements diagram. The questionnaires were submitted to the consumers and companies through two different Google Forms. The main significative results of these questionnaires were transformed into the specific requirements which will be inserted in the developed traceability model for the agri-food.
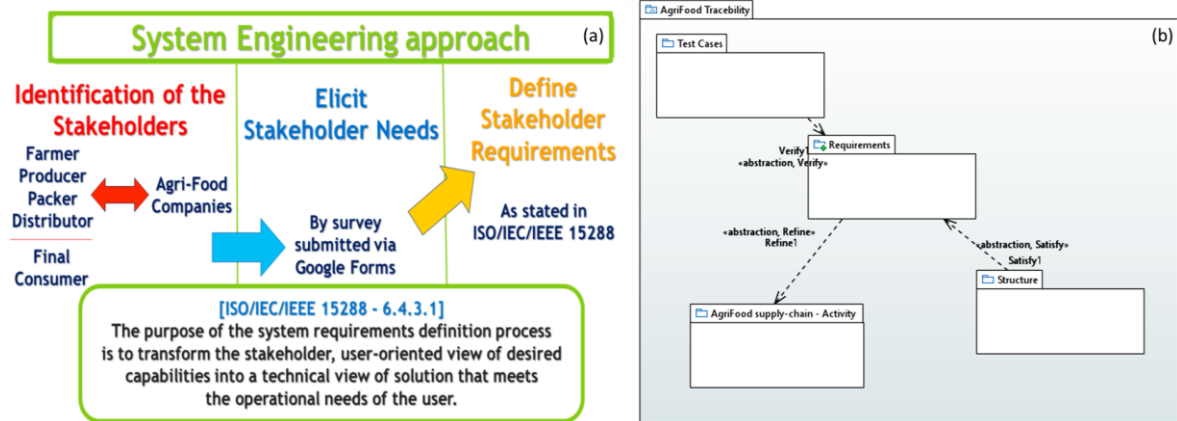
Figure 1: (a) The SE approach for the model development of the agri-food supply-chain about the traceability of information, (b) AgriFood Traceability Model Solution

3. Results and Discussion

The Requirement diagram is composed of: consumer requirements, business requirements, legislation requirements and environmental requirements. Due to the inherent parent-child relationship in the typical structure of the requirement package in SysML, it responds in a manner appropriate to the traceability of a system. In the Papyrus tool is also possible to draw up the table of requirements automatically (see Figure 2), which consents to better read, as in this case, a complicated and extended system of requirements.

	id	text
⊟ 📄 REQ-001 Requirements	REQ-001	*The system shall respond to this requirements set*
⊟ 📄 REQ-002 Consumer Requirements	REQ-002	The system shall be the set of these requirements defined by the consumers
REQ-006 expiration date	REQ-006	The system shall indicate the expiration date of the product
REQ-007 origin and provenance	REQ-007	The system shall indicate the origin and provenance of the product
REQ-008 list of ingredients	REQ-008	The system shall indicate the list of ingredients of the product
REQ-009 lot number	REQ-009	The system shall indicate the lot number of the product
REQ-010 nutritional table	REQ-010	The system shall indicate the nutritional table of the product
REQ-011 headquarters	REQ-011	The system shall indicate the headquarters production and packaging of the product
REQ-012 conservation mode	REQ-012	The system shall indicate the conservation mode of the product
REQ-013 net quantity	REQ-013	The system shall indicate the net quantity of the product
⊟ 📄 REQ-003 Business Requirements	REQ-003	The system shall be the set of these requirements defined by the companies
REQ-014 percentage increase in the co...	REQ-014	Companies consider acceptable a percentage increase in the cost of the product from 1% to 5%
⊟ 📄 REQ-004 Legislation Requirements	REQ-004	The system shall respond to the regulation derived from the article n.18 of EC/178/2002
REQ-015 stakeholders name	REQ-015	The name of each stakeholder shall be specified
REQ-016 products name	REQ-016	The name of each products - supplied and delivered in each steps - shall be specified
REQ-018 transactions time	REQ-018	The time of each transaction may be specified
REQ-017 transactions date	REQ-017	The date of each transaction shall be specified
REQ-019 products quantity	REQ-019	The quantity of each product shall be specified by volume or mass
⊟ 📄 REQ-005 Environmental Requirments	REQ-005	The system shall respond to these needs derived by the objectives of the next years
REQ-020 zero waste	REQ-020	Companies shall minimize the wastes and optimize the water resources

Figure 2: Requirements Table for the agri-food traceability system

3.1 Consumers' Needs and Requirements

The questionnaire submitted to the consumers received 160 answers and thus, they may be considered an acceptable sample to elicit needs and then define requirements. The answers are referred to a sample of people of different ages, predominantly residents of Calabria and more than half of them are employed with a fixed salary. People declared to spend a considerable part of their income on food shopping (see Figure 3a). It is really interesting to understand what are the needs of consumers of agri-food, because their needs could shift the interest of the market. The sample is homogeneous and representative of the Calabrian reality. In the future, it will be interesting to interview more people without a fixed salary, to make the proposed solution even more sustainable from a social point of view.

Besides the answers about the most important information to report on the label concerning the purchasing of food, all consumers' needs essentially refer to the need for *clarity of information*, *safety* and *quality*. From the answers, also given for every single food, there appears to be a real confusion on the part of the consumer, who indicates in the information to be reported, information that is already present, with the addition of a few specific

pieces of information on the specific food. Most of the information in Figure 4b is already part of the mandatory information to be reported according to European Regulation No. 1169/2011.
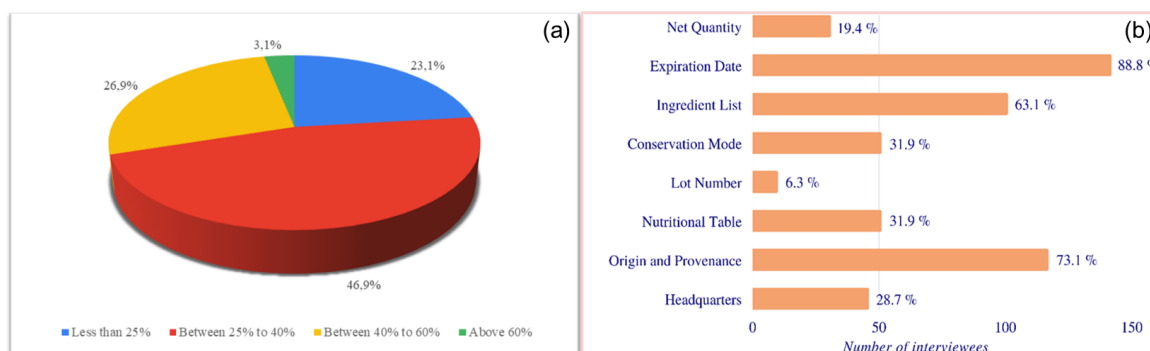

Figure 4: (a) Percentage of monthly income used in food shopping from the reference sample, (b) The most important information on the label for purchasing food

This means once again that the information, although already provided, is not clear to consumers. The clarity of information seems to be the greatest need of consumers. Being this information already mandatory, they may be easier to acquire; they could be considered quite general for each supply-chain. This information – *net quantity, expiration date, ingredient list, conservation mode, lot number, nutritional table, origin and provenance, headquarters* – could be considered as the consumers' requirements, because they are also measurable through the consumers' answers and therefore through the identified percentages. Each requirement can meet more than one need. In particular, clarity of information can be linked to net quantity, ingredient list, conservation mode, nutritional table and headquarters; safety can be linked to the expiration date, conservation mode, lot number and origin and provenance; the quality can be linked to net quantity and origin and provenance. Also, from the number of links between needs and requirements, clarity of information results to be the most important need, followed by safety and then quality.

3.2 Companies' Needs and Requirements

The questionnaire was submitted to some local companies through Google Forms, but it has obtained only eight answers; therefore, it cannot be considered as an acceptable statistical sample, but at the same time, the participating companies could provide some useful information about this study, also because they cover all three different stages of the supply-chain. The most important information was used as the companies' requirements and it is an indication of the maximum percentage increase acceptable in the cost of the product considered to improve the company traceability system. The indicated value is considered acceptable from 1% to 5%.

3.3 Other Requirements

Other possible requirements can be the requirements dictated by the laws and the environmental needs.
The Legislation Requirements are derived from article No.18 of European regulation No. 178/2002 about the standard of the European food legislation. It essentially consists of the declaration of the names of stakeholders and products, the transaction times and dates, and the quantity of products. Other requirements could come from other regulations about the specific product sector. Actually, in Italy, the voluntary agreements exist for: fresh and processed meat, milk, fish and fresh fruits and vegetables. The Environmental Requirements can also refer to legislation references, but in this case, general and specific environmental constraints of agri-food that are not included in the legislative ones were considered. In particular, one of the most important hypotheses of this work is zero waste.

3.4 AgriFood Traceability SysML Model Solution

The change of any requirement over time can be an opportunity to better model the system as a whole. Other changes identified when the system takes over a change in the behavior, that is when there is an unexpected event – e.g., tampering with a product – could be corrected and/or sanctioned almost immediately or in any case faster than now. The designed SysML modelling strategy in Papyrus considered the change of requirements over time, a constraint of the system: As a consequence, a high level of abstraction was considered in the SysML solution. The model – shown in Figure 1b – consists of four package diagrams, whose core is the requirement diagram. This last will act on the agri-food supply-chain, draw up an activity diagram and it will refine the supply-chain according to all requirements, which can be changed or updated at each time.

The activity diagram identified a generic and unified agri-food supply-chain, able to define each phase and each actor of the supply-chain: the production, the processing, the distribution, the waste management, the farmer, the producer, the packer, the distributor and the final consumer. The test cases package is used to verify the requirements, for each supply-chain, of which new information is available; for example, through the use of other tools, including other specific questionnaires to be submitted to the companies. The structure package is a parametric diagram. It expresses the constraints on the solution, starting from the constraints that express the properties of the identified traceability system. The structure consisted of different packages to express different constraints e.g., technical, performance, reliability, availability, power, mass, and cost. The constraints expressed from the technical solution will satisfy the requirements expressed by the system under consideration. The advantages of this SysML model also consist of a possible substitution or integration of the technical solution, if it represents an improvement in the traceability solution problem. For this reason, the structure consists of the generic characteristics of a traceability structure: identification, record, data link and report. In the identification block, two property classes were defined: the definition of the logistic units and the lot with the same production process. In this way, the actors and each phase of the supply-chain could be considered, having considered the integrated logistic concept as a starting point. Each logistic unit has multiplicity 1, because it must be unique. The record block consists of the definition and recording of the key information useful for traceability and chosen thanks to the requirements analysis. The choice to extend the information set depends on the specific test case – implemented in a test case diagram – represented by a specific supply-chain and eventual single company. From the test cases diagram the requirements are verified and also eventual update. In this case, the multiplicity value may be 0…*, which means that the information could be recorded many times. The update depends on the chosen technical solution. The data link block starts from the necessity to record the link between the products and the logistic units during all supply-chain. Therefore, in this class block, three properties were set and identified: the link between production batches, the link between batches and logistic units, and the link between logistic units. Each property has multiplicity 1, because it must be unique. The report block is based on two attributes: the common language and the shared method. Both these must be realized between all stakeholders of the supply-chain. The report must be readable to all. Both attributes are linked to the need for clarity of information – expressed by consumers and companies - and also from the technical solution. Papyrus can validate directly the designed model, without going into the complicated simulations, especially in the case of a high level of abstraction, which would be missing some implementation parameters necessary for the simulation of the technical phase. This type of validation consists of a better and faster check of what has been developed through project simulations. Being able to quickly test the behavior, one immediately realizes whether there are modelling and/or design errors or not.

4. Conclusions

Papyrus tool consents to make a basic simulation to understand the performance factors for the identified solution. Considering the high level of abstraction, for this AgriFood Traceability model, MBSE may help to trace requirements to implementation and verification artefacts in an automated way. Requirements represent the core of this AgriFood Traceability system and it can be updated at each time, from the specifications of the system under consideration and also, from the technological solution adopted to respond to the needs of the system. The AgriFood Traceability model has been designed at this high abstraction level and thanks to Papyrus its validity has been verified. The validation check phase did not highlight any errors or warnings. Thus, the idea to design a traceability system for the entire agri-food supply-chain, starting from the requirements of the system seems to be correct. Additional performance values of the model will be evaluated after the choice of the technical solution adopted for traceability.

References

Delligatti, L. (2014). *SysML distilled : a brief guide to the systems modeling language*.

Friedenthal, S., Moore, A., & Steiner, R. (2014). A Practical Guide to SysML: The Systems Modeling Language, Third Edition. *A Practical Guide to SysML: The Systems Modeling Language, Third Edition*, 1–606. https://doi.org/10.1016/C2013-0-14457-1

Ryan, M. J. (2013). 3.3.1 An Improved Taxonomy for Major Needs and Requirements Artifacts. *INCOSE International Symposium*, *1*. https://doi.org/10.1002/J.2334-5837.2013.TB03016.X

Stoewer, H. (2003). Modern Systems Engineering: A Driving Force for Industrial Competitivity! *Proceedings. 11th IEEE International Requirements Engineering Conference, 2003.*, 6–6. https://doi.org/10.1109/ICRE.2003.1232732

Technical Operations International Council on Systems Engineering (INCOSE). (2007).