

An Overview of Traceability: Towards a general multi-domain model

Kamal Souali*, Othmane Rahmaoui, Mohammed Ouzzif

ESTC, RITM Laboratory, Hassan II University ENSEM, Casablanca, Morocco

Email: kml.souali@gmail.com, othmane.rahmaoui@gmail.com, ouzzif@gmail.com

ARTICLE INFO

Article history:

Received: 03 April, 2017

Accepted: 02 May, 2017

Online: 17 May, 2017

Keywords:

Traceability

Trace

MDE

Model

Definition

ABSTRACT

Traceability for some people, is merely a tool to keep a history over something important that happened in the past. For others, it has no added value to their actual processes or products. In fact, it is becoming more and more valued. Traceability is still a vast area of research and an undiscovered field that if it is well used and managed, can provide a set of critical information or lead to something bigger. Many researches are still working to enhance its use and its integration by providing solutions to help users better manage and control their different elements (products, source code, documents, requirements, specifications, etc.). Nowadays, it is used in almost all domains as it can provide reliable information and helps improve efficiency and productivity. In this paper, we first present the state of the art on traceability and its use, through several examples. Then we provide a list of major techniques used in this field and propose our own traceability definition models.

1. Introduction

This paper is an extension of work originally presented in the 4th IEEE International Colloquium on Information Science and Technology [1] and was meant to show and explain the important role that traceability plays in different sectors.

Traceability, as defined in ISO (ISO 9001: 2000), is the ability to trace the history, application or location of that which is under consideration. D. Asioli, A. Boecker and M. Canavari say that it is not a new concept but a practice that we need to implement in order to comply with the standards and law rules [2]. Certainly, over the past few years, it has become a necessity in fields where the security or safety of consumers is questioned, especially in medical and food industries.

In software development also, this practice helps in the understanding, capturing, tracking and verification of software artifacts and their relationships and their dependencies during a software life-cycle [3]. As in [4], traceability was initially used to trace requirements from their source to implementation and test, when we talk about software development and now, it plays an

increasing role in defect management, change management and project management.

According to the Global Traceability Standard (GS1), Traceability Systems have become an integral part of doing business as they aim to identify and locate unsafe foods and validate the presence or absence of attributes that are important to consumers [5]. Even if they are not yet considered as a catalyst for financial gains, G.G.D. Nishantha, M.K. Wanniarachchige and S.N. Jehan say that they are able to ensure consumer trust, safety, reliability, accuracy and quality [6]. Its importance is reflected through its ability to solve issues and through its power to provide strong proofs or evidences.

These Systems, along with their ability to monitor the composition as well as the position of every lot in a supply chain, are seen as a powerful tool that is capable of defining new management objectives and improve the overall performance [7].

Our personal definition of traceability would be, the ability to keep a detailed history of all activities and changes that a particular object can undergo throughout its entire life cycle, taking into account the different relationships that may appear. This particular object can be a material, a product, a model or even a class in a software development platform.

*Corresponding Author: Kamal Souali ESTC, RITM Laboratory, Hassan II University ENSEM, Casablanca, Morocco | Email: kml.souali@gmail.com

Traceability, if it is used in the right way, can provide a set of critical pieces of information such as the source, the destination, the location, the time, the link, in addition to the actors that were involved in the whole process. As in [8], ubiquitous traceability is achieved automatically, as a result of collecting, analysing, and processing every piece of evidence from which trace data can be inferred and managed.

The remainder of this paper is organized as follows: In Section 2, we will provide a set of definitions extracted from two major sectors, namely the food industry and information technology. Section 3 presents examples of traceability uses in different areas. A list of major techniques that were used to enhance the traceability will be the object of Section 4. In Section 5, we propose a set of definition models related to previously mentioned sectors. We will discuss the proposed traceability definition model in Section 6 and we will give a brief conclusion in Section 7.

2. Definitions

Traceability management is the planning, organization, and coordination of all activities related to traceability, including the creation, maintenance and use of trace links [9], not only in software development, but also in our daily life (e.g. memorizing events, tasks, activities, etc.).

In any area or sector, the definition of "traceability" is based on a number of criteria and limitations according to the used law or standards like the European General Food Law (EGFL) and the GS1 or simply, describes its purpose in a specific context.

Authors in [10] stated that there is no exact, single definition of traceability and that it has a large number of different meanings, which depend on the industry sector, on the supply chain, and on the perspectives of both the suppliers and the users of such information. However, we intend in section 5, to prove that a common definition can be established by means of models.

2.1. Food Industry

The EGFL defines traceability as the ability to trace and follow a food, feed, food-producing animal or substance through all stages of production and distribution.

According to A. F. Bollen and J.P. Emond, traceability is a well-coordinated and a well-documented movement of product and documented activities associated with the product, from producer, through a chain of intermediaries, to the final consumer [10].

M. Gooch and B. Sterling say that it is the ability to follow an item, or a group of items (whether animal, plant, food product, or ingredient) from one point in the value chain to another, either backwards or forwards [11]. Thus, food chain traceability goes from raw materials to consumption. This is almost the same definition given by F. Dabbene, P. Gay and C. Tortia, as they assume that products "moving" along the Food Supply Chain (FSC) are both tracked and traced [7].

Tracking is the process by which a product is followed from upstream to downstream in the Supply Chain. Tracing is the reverse process of tracking. The tracing process tends to reconstruct the history of a product through the information recorded in each step of the Supply Chain, identifying the source

of a food or group of ingredients and consequently the real origin of a product [12]. These two primary functions of traceability are known as Trace-Back and Trace-Forward, as the movement can be traced one step backwards and one step forward at any point in the supply chain [6].

As in [13], traceability can either be internal or external. Internal traceability is within one company and relates to data about raw materials. While external traceability focuses on the product information from one link in the chain to the next (tracking a product batch and its history through the entire production chain).

In the food industry, traceability requires that each lot or amount or batch of food material is given a unique identifier which accompanies it and is recorded at all the stages of its progress through its food chain [14].

J. C.C. Martins and R. J. Machado said that a traceability system must record and follow the trail, since products that come from suppliers, are processed and distributed as end products [15]. The traceability presented by these records must contain a set of reliable pieces of information in order to ensure the minimum requirements. In fact, as stated in [11], it has three key essential information components: (1) identification of product attributes, (2) identification of premises and (3) identification of movement.

In the same context, P. Olsen and M. Borit have carried out an insightful comparative study of existing definitions [16]. By combining the best parts of these definitions, they concluded by saying that the simplest yet the most complete definition of traceability is the ability to access any or all information relating to that which is under consideration, throughout its entire life cycle, by means of recorded identifications.

2.2. Information Technology

In the field of software engineering, the IEEE Standard Glossary of Software Engineering Terminology defines traceability as the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another [17].

It is the ability to inter-relate any uniquely identifiable software engineering artefact to any other, to maintain the required links over time, and to use the resulting network to answer questions of both the software product and its development process [8]. It is a key element of any rigorous software development process that, provides critical support for many development activities [18].

When we talk about traceability in software development, we often refer to Requirement Traceability, which is an activity that allows creating links between and within software artefacts [19]. The definition of Requirement Traceability (RT), according to O. C. Z. Gotel and A. C. W. Finkelstein, is the ability to describe and follow the life of a requirement, in both a forward and backward direction [20]. Other definitions can be purpose-driven, solution-driven, information-driven or direction-driven.

These authors specify that there are two types of RT: pre-requirements specification traceability (Pre-RST), which is concerned with those aspects of a requirement's life prior to its inclusion in the Requirement Specification (RS), and post-requirements specification traceability (Post-RST), which is concerned with those aspects of a requirement's life that result from its inclusion in the RS.

More details about software traceability were listed in [8], including seven research areas and their associated directions which must be addressed in order to achieve ubiquitous traceability.

3. Uses of Traceability

In the food industry, it is considered as a mechanism used to keep the history of a raw or semi-finished unit during manufacturing and until this unit is delivered. It has a great potential to improve food safety as well as to promote consumer protection, by providing quality information [21].

In the field of Information Technology, traceability is used to list all activities of an entity on a system in execution. An example of such is the use of recovery logs or event logs in some cases. As R. Clayton explained, it is the ability to track down the originator of an action (seen as the flip side idea to “anonymity”) and attempts to identify the IP address that caused an action to occur [22]. For instance, Law Enforcement Agencies (LEA) can use traceability to detect “Hi-Tech” crimes through data retention (causing logs to be preserved for a known period) and data preservation (ensuring that logs of special interest are not destroyed).

It is also used to clearly identify the sources behind some statistical analysis. Authors in [23] stated that it is the property which enables the understanding of where the analysis data come from and facilitates transparency. They have proposed a set of traceability pairs (relation criteria and factors) to define all the variables required in an analysis and hence establish the link between the final result and all the sources used. Moreover, traceability can strengthen the link between the requirements put in place, the specifications and the artefacts throughout the phases of a software development, using Requirements Traceability Matrix [19].

As authors in [9] explained in details, it allows creating and using links between software artefacts, which for example allows to connect the origin of a requirement with its specification or development artefacts to each other throughout the software lifecycle. These connections are called trace links, and link a source artefact to a target artefact. These artefacts can be of different types, such as a requirement, a model element, a line of code, or a test case.

In aerospace industry [24], traceability can be used to find the design related causes if a product does not function as expected. It is provided by establishing the relations between the design data and the requirements together with the relations between the components and the identifiers.

In Supply Chain Management (SCM), R.R. Pant, G. Prakash and J. A. Farooque, traceability is defined in terms of what, how, where, why and when aspects of underlying product along a supply chain [25].

In logistics, traceability may be used to optimize routes and improve planning and management. It may also work with accounting applications to evaluate inventory or with controlling applications to identify process inefficiencies [15].

In electronics, traceability is used to keep track of all information related to changes and transformations which are applied to identified Printed Circuit Board (PCB) or other electronic components. Starting from the original batches and sources, this information is mainly, the Bill of Materials (BOM),

the measurements, the list of operations in the process chain and the final destinations to whom or where the boards must be shipped. As stated in [26], it is required for fulfilment of safety standards such as ISO 26262.

Furthermore, traceability is also used in biology. An example of such, is to trace Genetically Modified (GM) animals that may similarly yield improvements in animal breeding, genetics and reproduction [27].

4. Techniques

In order to help users better manage their traceable items, the traceability mechanism has been enhanced by making use of different approaches that vary from using simple information retrieval techniques to the use of ontologies, graphs or even models.

4.1. Information Retrieval

A. De Lucia, A. Marcus, R. Oliveto and D. Poshyanyk explained that Information Retrieval based methods or techniques like probabilistic, vector space and Latent Semantic Indexing models are used to recover traceability links on the basis of the similarity between the text contained in the software artefacts [28]. The higher the textual similarity between two artefacts is, the higher the likelihood that a link exists between them.

As in [29], this approach focuses on automating the generation of traceability links by similarity comparison between two types of artefacts.

4.2. Ontology

J. C.C. Martins and R. J. Machado proposed the use of software engineering methods and techniques to aggregate, disambiguate and blend existing knowledge [15]. They have used ontologies as a requirements modeling technique and developed specific traceability taxonomy in order to pursue the continuous improvement and answer the requirements of increased efficiency by tracking manufacturing activities information.

S. Bendriss and A. Benabdelhafid used DAML-S which is a generic ontology that can be applied in all areas [30]. They have adapted it by integrating and adding their specific ontology “Product Traceability Service”, which describes all the web services of their traceability system. These services are dedicated to supply chain.

4.3. Graphs

As detailed in [19], “TraceMe” is an Eclipse module-based plug-in, that can be used to capture and maintain traceability links between different types of artefacts. According to the authors, this plug-in allows the software engineer to define different artefacts categories, capture traceability links between the defined artefacts categories and manage the traceability information through XML files. Traceability dependencies (trace links) are then displayed as graph.

Other researchers tend to use graph-based techniques in order to create trace links of test case scenarios and therefore, enhance the test coverage measurement and analysis [29].

Additionally, there are other plug-ins in the internet which are capable of tracing issues to both requirements and tests and creating the related traceability matrix.

4.4. Models

According to N. Sannier and B. Baudry, domain-specific modeling, which offers the capability to manipulate business domain concepts and traceability modeling, are Model-Driven Engineering (MDE) techniques that could address various aspects of requirement's formalization [31]. These authors proposed to combine both MDE and Information Retrieval (IR) techniques to improve requirements organization and traceability while handling textual ambiguous requirements documents.

MDE gives the basic principles for the use of models as primary artefacts throughout the software development phases and presents characteristics which simplify the engineering of software in various domains, such as Enterprise Computing Systems. A model is a symbolic system expressed in a language and each kind of model is represented by an appropriate modeling language and can be applied to certain purposes [3].

M. Thakur, B. J. Martens and C. R. Hurburgha defined a data model as a coherent representation of objects from a part of reality [32]. They used the modeling technique to create a database model capable of recording all the transformations related to incoming and outgoing grain lots, as well as the transformations that take place internally in the whole supply chain.

By making use of modeling techniques, C. Szabo and Y. Chen proposed "SeMMA" (Semantic Multi-Modeling Architecture), which is a multi-modeling architecture that permits the semantic integration of models defined in various languages, and ensures multi-model consistency when changes across different models occur and relies on three main modules, namely, the Change Analyzer Module, the Consistency Checker Module and the Warning Module [33].

In the same context, S. Bendriss and A. Benabdelhafid proposed a product data model which takes into account the different elements necessary for traceability, namely, the product in its various states, the various operations on the product, the occurred events, the resources used and the spatiotemporal location of the product [30].

In software development, requirement traceability can be described as a feature model to define a product [29]. It consists of a graph with features as nodes and feature relations as edges. If the number of features is very high, then the representation of features and their relations are displayed by tables.

On the other hand, the authors in [18] presented an approach on how to build a multi-domain traceability framework. It consists of defining first a Traceability Information Model (TIM) which represents the core element of any traceability framework (artefacts/relations) and may refer to artefacts (documents, models, databases, project activities context) from different domains, then deriving traceability information from sources, record the information in a Traceability Model (TM) and finally, performing traceability analyses, based on traceability goals.

Another example of such technique is presented in [26], where authors proposed an Eclipse plugin which uses the Eclipse Modelling Framework (EMF) as its base technology and stores the traceability model as an EMF model. This tool helps both users and project managers to create, customize and maintain traceability links, whose types depend on the company, development context and process used.

For more details about MDE techniques, Galvão and A. Goknil have listed many traceability approaches in MDE and evaluated them using five comparison criteria: representation, mapping, scalability, change impact analysis and tool support [3]. They classified these approaches into three categories: requirements-driven approaches, modeling approaches and transformation approaches.

4.5. Others

F. Furtado and A. Zisman proposed a new traceability approach called "Trace++", a traceability technique that extends traditional traceability relationships to support the transition from traditional to agile software development [34]. This technique extends the use of information sets and consists of six elements: the agile related problem, the trace relations, a set of all source artefacts, a set of all target artefacts, a set of additional information and finally, the type of relations.

Other techniques tend to use XML as the main tool to represent models and trace links.

As stated in [29], these techniques are classified as Hypertext-Based techniques. But there are others which can be either Rule based, Event based, Value-based or Scenario-based.

5. Definition Models

As stated before, there is no single or unique definition for traceability, since the term is described according to both its context and its purpose. Based on this and on the elements extracted from the other definitions, we hereby propose a definition model for the main sectors.

In Food Industry, the purpose of traceability is to trace the initial product with the raw materials from the start, till the very end of the production chain. Figure 1 represents the internal traceability in this field, where p is the initial item, P is the final product and $i_0, i_1 \dots i_n$ are the set of information that describes the movement from one point to another.

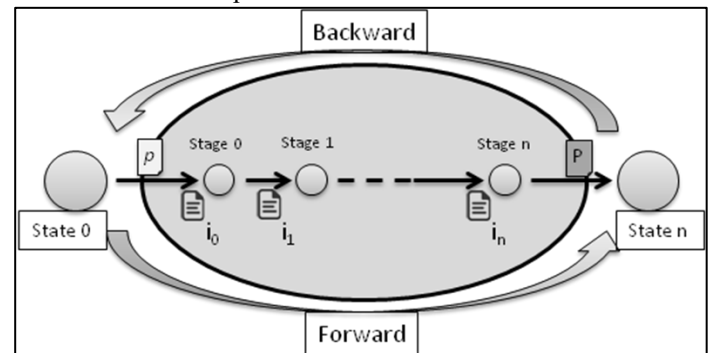


Figure 1: Traceability in Industry

By simply adjusting these elements, this representation can also describe the traceability in the Supply Chain or Logistics, tracing lots from the warehouse to every destination of the distribution chain.

Furthermore, it is only when two or more separate representations of this size are connected, that we basically speak about external traceability. Otherwise, it is still internal.

In the field of Information Technology, one example use of traceability is to create links between customer requirements or

specifications and the supplier software. As shown in Figure 2, s refers to the initial specification (requirement) or source, while O refers to the final object. The relation between the different stages of development is represented by $r_0, r_1 \dots r_n$.

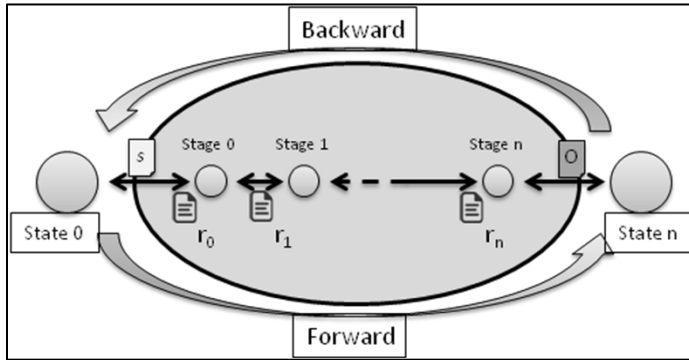


Figure 2: Traceability in Software Development

When this definition is applied to the model driven engineering (Figure 3), relations are replaced by a set of transformations $t_0, t_1 \dots t_n$ between predecessors and successors, representing the same system S . Elements s and O will be replaced respectively by m for the initial model and M the final model. Not to mention that each transformation can be represented likewise.

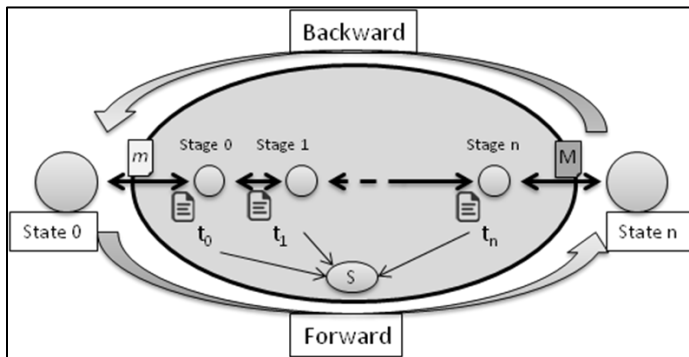


Figure 3: Traceability in Model Transformation

To sum up, we can say that the three proposals have a set of elements in common:

- Items: units that need to be traced and followed.
- Stages: positions where the units are processed.
- Relations: links between predecessors and successors.
- Activities: set of processes that were applied to the units

These elements will lead us to set a common model for traceability, which will be the basis of our future studies.

6. Discussion

Traceability in our point of view, as stated before, is the ability to keep a detailed history of all activities and changes that a particular object can undergo throughout its entire life cycle, taking into account the different relationships that may appear. It can be internal or external and can be used in two different ways either forward or backward.

As presented in the previous section, every traceable item is moving from an initial state to a final state, through numerous stages. In each stage, the output is the result of an activity that takes

into account inputs from the previous stage and keeps the link to the origin. At the end, and since inputs and outputs are interrelated, tracing forward and backward is possible.

Thus, we can combine these facts to establish a common definition model that can be used to define “Traceability” everywhere (Figure 4).

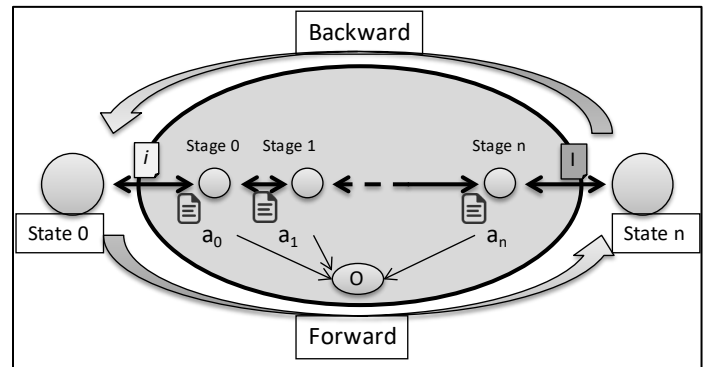


Figure 4: Generic Traceability Model

Here, i refers to the initial traceable item, where f is the final traceable item. The activities that the traceable item undergoes are represented by $a_0, a_1 \dots a_n$. O is the origin or the representation of all original characteristics of the traceable item. These characteristics do not change and are only updated if a new property was discovered when moving from one stage to another.

Certainly, once this generic model is deployed on a particular platform, it will be subjected to a large amount of data, of different types. Hence, it is mandatory to consider the following challenges:

- How to address the problem of time vs Big Data during information access?
- How can we manage to order the accessed traceability information by degrees of priority or importance?

We intend on enhancing our model by adding a set of rules and other traceability related properties.

For instance, a “weight” or a “priority” measure can be introduced and assigned to each traceability information, after the classification process, or we can improve the representation of trace links by including additional factors. Thus, only the most important set of information is shown when tracing an item, either backward or forward.

7. Conclusion

Traceability can ensure quality, safety, reliability and accuracy. Furthermore, it can help companies improve productivity, reduce costs and gain consumer’s trust.

According to GS1 [5], traceability may assess other business systems and tools such as quality management, risk management, information management, logistical flows, commercial advantage and evaluation of management demands.

In this paper, we have listed recent definitions related to traceability from two major sectors. We have presented definition models for these sectors and proposed our generic traceability

model, by combining a set of common elements, which stands as the basis of our research. In the same context, we have listed also, the uses and purposes of traceability as well as the major techniques applied in this research field.

In future work, we intend to refine our model with new elements and then, deploy it and use it in E-learning environments. Furthermore, a study will be initiated to decide whether or not this model can be applied to other fields or needs further enhancements. The final purpose will be to implement a general model, able to satisfy all traceability needs and requirements.

Conflict of Interest

The authors declare no conflict of interest.

References

- [1] K. Souali, O. Rahmaoui, M. Ouzzif, "An overview of traceability : definitions and techniques", 4th IEEE International Colloquium on Information Science and Technology (CiSt'16), 2016.
- [2] D. Asili, A. Boecker, M. Canavari, "Perceived traceability costs and benefits in the Italian fisheries supply chain", *International Journal on Food System Dynamics* 2(4), pp 357-375, 2011.
- [3] I. Galvão, A. Goknil, "Survey of traceability approaches in Model-Driven Engineering", 11th IEEE International on Enterprise Distributed Object Computing Conference (EDOC), 2007.
- [4] G. Regan, F. McCaffery, K. Mc Daid, D. Flood, "The barriers to traceability and their potential solutions: towards a reference framework", 38th Euromicro Conference on Software Engineering and Advanced Applications, 2012.
- [5] GS1 Global Traceability Standard Business Process and System Requirements for Full Chain Traceability, <http://www.gs1.org/traceability/gts>, 2007.
- [6] G.G.D. Nishantha, M.K. Wanniarachchige, S.N. Jehan, "A pragmatic approach to traceability in food supply chains", 10th International Conference on Advanced Communication Technology (ICACT 10), 2010.
- [7] F. Dabbene, P. Gay, C. Tortia, "Traceability issues in food supply chain management: A review", *Biosystems Engineering* vol. 120, pp 65–80, 2014.
- [8] J. Cleland-Huang, O. Gotel, J. H. Hayes, P. Mäder, A. Zisman, "Software traceability: trends and future directions", The 36th International Conference on Software Engineering, 2014.
- [9] R. Wohlrab, J. P. Steghöfer, E. Knauss, S. Maro, A. Anjorin, "Collaborative Traceability Management: Challenges and Opportunities", The IEEE 24th International Requirements Engineering Conference, 2016.
- [10] A. F. Bollen and J.P. Emond, "Traceability in postharvest systems", *Postharvest Handling*, pp.485-504, 2014.
- [11] M. Gooch, B. Sterling, "Competitive advantage of food traceability to value chain management", <http://vcm-international.com/>, 2013.
- [12] T. Pizzuti, G. Mirabelli, M. A. Sanz-Bobi, F. Gómez-González, "Food track & trace ontology for helping the food traceability control", *Journal of Food Engineering* vol. 120, pp 17–30, 2014.
- [13] J. T. Mgonja, P. Luning, J. G.A.J. Van der Vorst, "Diagnostic model for assessing traceability system performance in fish processing plants", *Journal of Food Engineering* vol. 118, pp 188-197, 2013.
- [14] B. Welt, J. R. Blanchfield, International Union of Food Science And Technology, IUFOST Scientific Information Bulletin (SIB), Food Traceability, 2012.
- [15] J. C.C. Martins, R. J. Machado, "Ontologies for product and process traceability at manufacturing organizations: a software requirements approach", Eighth International Conference on the Quality of Information and Communications Technology, 2012.
- [16] P. Olsen, M. Borit, "How to define traceability", *Trends in food science & technology* vol. 29, pp 142-150, 2013.
- [17] IEEE Standards Software Engineering, IEEE Standard Glossary of Software Engineering Terminology, IEEE Std. 610-1990.
- [18] M. Taromirad, N. Matragkas, R. F. Paige, "Towards a multi-domain model-driven traceability approach", The 7th Workshop on Multi-Paradigm Modeling, 2013.
- [19] G. Bavota, L. Colangelo, A. De Lucia, S. Fusco, R. Oliveto, A. Panichella, "TraceME: Traceability Management in Eclipse", 28th IEEE International Conference on Software Maintenance (ICSM), 2012.
- [20] O. C. Z. Gotel, A. C. W. Finkelstein, "An analysis of the requirements traceability problem", *Proceedings of the First International Conference on Requirements Engineering*, 1994.
- [21] M. Garcia Martinez, F. M. Brofman Epelbaum, "The role of traceability in restoring consumer trust in food chains", *Food Chain Integrity*, 2011.
- [22] R. Clayton, "The limits of traceability", from http://www.cl.cam.ac.uk/~rnc1/The_Limits_of_Traceability.pdf, 2000.
- [23] S. Zhu, L. Yan, "Methods of building traceability for ADaM data", *Proceedings of the PharmaSUG2011 Conference - Paper CD05*, 2011.
- [24] E. Tekin, "A method for traceability and as-built product structure in aerospace industry", *Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 2014.
- [25] R.R. Pant, G. Prakash, J. A. Farooque, "A framework for traceability and transparency in the dairy supply chain networks", XVIII Annual International Conference of the Society of Operations Management (SOM-14), *Procedia - Social and Behavioral Sciences* vol. 189, pp 385 - 394, 2015.
- [26] S. Maro, J. P. Steghöfer, "Capra: A Configurable and Extendable Traceability Management Tool", IEEE 24th International Requirements Engineering Conference, 2016.
- [27] A. Lievens, M. Petrillo, M. Querci, A. Patak, "Genetically modified animals: options and issues for traceability and enforcement", *Trends in Food Science & Technology* vol. 44, pp 159-176, 2015.
- [28] A. De Lucia, A. Marcus, R. Oliveto, D. Poshyanyk, "Information retrieval methods for automated traceability recovery", *Software and Systems Traceability*, pp 71-98, 2011.
- [29] M. Shahid, S. Ibrahim, H. Selamat, "Test coverage measurement and analysis on the basis of software traceability approaches", *International Journal of Information and Electronics Engineering*, Vol. 1, No. 2, pp 115-119, 2011.
- [30] S. Bendriss, A. Benabdelhafid, "Enabling product traceability through data modeling and semantic web service ontologies", *International Conference on Advanced Logistics and Transport (ICALT)*, 2013.
- [31] N. Sannier, B. Baudry, "Toward multilevel textual requirements traceability using model-driven engineering and information retrieval", *Model-Driven Requirements Engineering Workshop (MoDRE)*, IEEE, 2012.
- [32] M. Thakur, B. J. Martens, C. R. Hurburgha, "Data modeling to facilitate internal traceability at a grain elevator", *Computers and Electronics in Agriculture* vol. 75, pp 327-336, 2011.
- [33] C. Szabo, Y. Chen, "A model-driven approach for ensuring change traceability and multi-model consistency", The 22nd Australian Conference on Software Engineering, 2013.
- [34] F. Furtado, A. Sizman, "Trace++: A Traceability Approach to Support Transitioning to Agile Software Engineering", The IEEE 24th International Requirements Engineering Conference, 2016.