# Requirements Traceability Related Work

In early days, the value of requirements traceability practice only aware in safety critical systems. In practise, requirements traceability is crucial to monitor whether system design and implementation covered all the features defined in the requirements specification or not. System verification, validation and change impact analysis are the key components addressed by requirements traceability. Recently, the rapid development of computer-added software application system in many sectors broaden the needs of requirement traceability in various domains. Thus, industrial practitioners and researchers are constantly exploring the standard framework and technique to integrate the traceability principal in various context. Primary goal of traceability is to maintain the association between requirements and its project artefacts including, design specification, source code and test cases in both forward and backward directions throughout the software development life cycle (Gotel & Finkelstein 1994).

In (Cleland-Huang, Chang & Christensen 2003) paper, the researchers introduced Event Based Traceability (EBT) approach to handle the subsequent change requirements request in software development projects. In EBT approach, the researchers used the concept of events and notification services to develop dynamic traceability links rather than establishing static requirement links. In architecture wise, EBT contains three main components namely, requirement manager, event server and subscriber manager. Requirement manager task include monitor requirements, change request and publishing change request notification to event server. Event manager service is responsible for listening event notification from requirement manager and forwarding the notifications to corresponding subscribers. At Last, subscriber manager monitor the event notification forwarded by event manager and then, decide to store incoming events either in events log for human review or process automatically based on a set of defined rules, depending on event and subscriber type. The study analysis the effectiveness of EBT approach by conducting case-studies approach.

In later year, (Cleland-Huang, Zemont & Lukasik 2004) research proposed trace strategies solution called Traceability for Complex Systems (TraCS) to maximize the traceability effort in terms of projects return-on-investment (ROI) requirements. The paper identified the gap of strategic decision requirements in designing traceability for the system. The study indicates that although most organization implemented the traceability practice in their system initially, the ongoing cost of maintaining the traceability links holding them back from utilizing it in the later stage of the project. In TraCs system, it first categorized the requirements using the risk analysis technique. Based on the risk analysis report, the project teams decide which requirements are needed to trace links throughout the development life cycle. With this approach, it minimize the traceability maintenance cost in one hand and improve the traceability effort on the other hand.

In (2006), Ozkaya and Akin realized the needs of computer-aided requirements traceability tool to manage the association between initial requirements specification and prototype form design exploration process. The idea behind the application is integrate requirements traceability concept in prototyping phase. The study presented the prototyping tool called DesignTrack, which embedded requirements traceability features in designing prototype model. The researcher applied the Computational Hybrid Assistance for Requirement Management (CHARM) process framework presented in their earlier work (Ozkaya 2005) to develop DesignTrack tool. In DesignTrack application, it provides an interactive medium for architects and analysts to explore their prototype form design along with initial requirements specification in traceable way. The relationships between requirements and prototype designs can be builded along with design exploration to manage the requirements management tasks. The study applied Leadership in Energy and Environment Design (LEED) requirements guidelines to evaluate the application(Ozkaya & Akin 2006).

Valderas and Pelechano (2009) applied requirements traceability principle in the context of model-driven web application system. The researcher created a tool called TaskTracer to generate traceability report to assist analysts to monitor how the conceptual models are structured to support requirements. In (Mellor, Clark & Futagami 2003) paper, it pointed out the popularity of model-driven development in software projects nowadays. According to the paper, software systems which are developed based on the programming language such as (C#, Java and SmallTalk) can be simply classified as model-driven development (MDD) projects. In MDD, the project phases are divided into three components; namely, requirements model, conceptual models and code. In requirement model, it study and define user needs. Then, requirements models is transform into one or more conceptual models, which produce the system process excluding technological aspects. In final, these models are either directly covert into code, if it contains enough information or transform into other models to cover the needs technological concepts to convert into code (Valderas & Pelechano 2009).

Similarly in (Zhang et al. 2010) paper, the researchers introduced the traceability concept in the area of agribusiness production system. The paper proposed the evolutionary prototyping model approach to study the current business process in order to develop effective traceability system. The study conducted the case study in the meat production firm located in Shandong province, China. In the paper, the researchers noted that when considering traceability aspects of the system, it is important to take into account ‘requirements are not static, but dynamic’(Zhang et al. 2010, p. 1556). Hence, the usefulness of traceability system relies on how well the system can handle the frequently changing needs of business.

Recently, (Regan et al. 2013) developed a new traceability process accessment model called Med-Trace to apply in medical device software system. Alike other safety critical systems, developing medical devices must comply with defined standards and regulatory guildelines to be able to distribute the product in the market. According to the paper, the challenging part of intergrating traceability in medical context is the lack of detailed guildelines to implement the practice in risk management process. Thus, the researcher presented the lightwieight assessment method to analysis the current state of medical device software traceability weakness and develop the improvement pathway. In Med-Trace assessment method, the process is carried out internally within the oranganization in an iterative way by taking assessors feedback reports as input for future improvement process. The researchers evaluated their method on two software-based medical devices systems by Small to Medium Sized (SME) companies located in Ireland and UK. To date, Med-Trace model is actively practising in these two companies.

# References

Cleland-Huang, J., Chang, C.K. & Christensen, M. 2003, 'Event-based traceability for managing evolutionary change', *IEEE Transactions on Software Engineering*, vol. 29, no. 9,pp. 796-810.

Cleland-Huang, J., Zemont, G. & Lukasik, W. 2004, 'A heterogeneous solution for improving the return on investment of requirements traceability', *Proceedings. 12th IEEE International Requirements Engineering Conference, 2004.*, pp. 230-9.

Gotel, O.C.Z. & Finkelstein, C.W. 1994, 'An analysis of the requirements traceability problem', *Proceedings of IEEE International Conference on Requirements Engineering*, pp. 94-101.

Mellor, S.J., Clark, A.N. & Futagami, T. 2003, 'Model-driven development - Guest editor's introduction', *IEEE Software*, vol. 20, no. 5,pp. 14-8.

Ozkaya, I. 2005, 'Requirement -driven design: Assistance for information traceability in design computing',Carnegie Mellon University, ProQuest Dissertations Publishing.

Ozkaya, I. & Akin, Ö. 2006, 'Tool support for computer-aided requirement traceability in architectural design: The case of DesignTrack', *Automation in Construction*, vol. 16, no. 5,pp. 674-84.

Regan, G., Mc Caffery, F., Mc Daid, K. & Flood, D. 2013, 'Medical device standards' requirements for traceability during the software development lifecycle and implementation of a traceability assessment model', *Computer Standards & Interfaces*, vol. 36, no. 1,pp. 3-9.

Valderas, P. & Pelechano, V. 2009, 'Introducing requirements traceability support in model-driven development of web applications', *Information and Software Technology*, vol. 51, no. 4,pp. 749-68.

Zhang, X., Lv, S., Xu, M. & Mu, W. 2010, 'Applying evolutionary prototyping model for eliciting system requirement of meat traceability at agribusiness level', *Food Control*, vol. 21, no. 11,pp. 1556-62.