

# Visualization representing benefits of pre-requirement specification traceability

Sohaib Altaf <sup>1</sup>, Asadullah Shah\*<sup>2</sup>, Najma Imtiaz <sup>2</sup>, Abdul Salaam Shah <sup>3</sup>, Syed Faiz Ahmed <sup>3</sup>

<sup>1</sup>Foundation University Rawalpindi Campus, Pakistan

<sup>2</sup>International Islamic University Malaysia

<sup>3</sup>Universiti Kuala Lumpur

\*Corresponding author E-mail: asadullah@iium.edu.my

## Abstract

Representation of Pre-RST information is very useful using visualized elements for realization of benefits of requirement traceability. This improves the practitioner motivation to maintain Pre-RST information during life cycle processes. Few researchers proposed visualization for Post-RST due to which many benefits of requirement traceability cannot be realized. This paper proposed an improved visualization representing Pre-RST information that demonstrates various benefit of requirement traceability. In order to evaluate empirically, an experiment is conducted and textual representation of traceability information is obtained. In order to strengthen our claim a survey is conducted to compare textual representation of traceability information with proposed visualization and results are compiled.

**Keywords:** Pre-Requirement Specification Traceability Pre-RST, Requirement Traceability, Software Requirement Specification SRS, Benefit of Traceability, Requirement Visualization, Value Based Traceability

## 1. Introduction

Software Traceability has many benefits throughout the software development lifecycle processes i.e. pre-requirements, requirements, architectural, code, test, maintenance processes etc. These benefits are obtained in term of conformity of quality, artefacts re-usability, verifiability, satisfaction, maturity, confidence, changeability, maintenance, usability and reliability [2, 7, 11, 16, 17, 32, 30]. Its importance is commonly accepted by many organizations, regulatory bodies and standardizing institutions [17, 32, 33].

Requirement engineering process is most critical process among all other life cycle processes. Software Traceability plays an important role in developing and managing requirements. It may assists in understanding requirement, re-use, capture history, tracking, validation as well as it is also useful to audit the performance of analyst [2, 16, 18, 33] and conforming requirements on quality measures [2, 33, 30]. It is most important to capture and preserve traceable information in order to maintain traceability among requirements, from requirements to its sources in backward direction and from requirement to design, code and rest of life cycle artefacts in forward direction.

Lot of traceability techniques are proposed [5, 6, 7, 8, 9, 10, 11, 12, 17] as well different tools offer various technology oriented solutions, are available now to automatically capture the traceability information [15, 21, 42], however many benefits of requirement traceability are yet not achieved [15, 18, 20, 30, 32, 35]. These tools integrate various visualization elements that show associations between different requirement artefacts in order to achieve benefits of traceability. However yet they are representing limited notations and metaphors for requirement traceability. Many useful concepts from traceable information can be mapped to various visualized metaphor enhancing visibility and perceptual ability. Such a visualized structure can enhance various decisions during

life cycle processes due to activation of pre-attentive inputs and less involvement of cognitive abilities [2, 18, 20].

The focus of this paper is on addressing Pre-RST issues and integrating pre-attentive elements that can represents cluster of traceable links in various pictorial concepts and provide good insight in order to assist requirement analyst in useful decisions. Among traceability problems, there is yet little research on pre-requirement specification traceability Pre-RST issues which makes this area fertile for research [4, 5, 6, 7, 8, 9, 10, 11, 12 and 17]. Valuable contributions have been made by researchers in area of requirement traceability for almost more than two decades [2, 4, 14, 15, 18, 20]. Most of these contributions are in area of traceability techniques, tools, visualization etc. that belongs to Post-RST [2, 3, 14, 15, 40, 34, 21]. Many problems of traceability arise due to difficulties in maintaining Pre-RST and survey reported various challenges related to Pre-RST. Maintaining Pre-RST is gaining significant importance in Requirement Engineering.

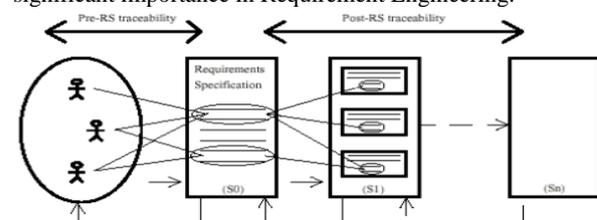


Fig. 1: Pictorial example of Pre-RST and Post-RST [16]

### 1.1. Definition of post- and pre-requirement specification traceability

Definition of requirement traceability is comprehensively discussed in various publications [16, 17, 32, 30, 36, 37, 47]. Gotel and Finkelstein [16] define requirement traceability as "Requirements traceability refers to the ability to describe and follow the life of a requirement, in both a forwards and backwards direction

(i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases)". They suggest two basic types as shown in fig 1; one is Pre Requirement Specification Traceability Pre-RST that "is concerned with those aspects of a requirement's life prior to its inclusion in the Software Requirement Specification SRS (requirement production)"; whereas other is Post Requirement Specification Traceability Post-RST that "is concerned with those aspects of a requirement's life that result from its inclusion in the SRS (requirement deployment)". "In software and systems engineering, a trace is a specified triplet of elements comprising: a source artefacts, a target artifact and a link associating the two artifacts" [47].

Pre-RST refers to various source, intermediate and target artefacts that involves in development of requirements. These artefacts represent both finished and unfinished artefacts as shown in fig 2. Finished artefacts i.e. User Stories, Goals, Use Cases, Vision Document, SRS etc evolve for on-going refinement in iterations with some version control process [33, 30, 35]. Requirement knowledge is organized in different sections in finished artefacts and its different versions show its evolution history whereas different association with other finished artifacts shows dependency either it is derived or served as input to other finished artefacts [16, 17, 18, 30].

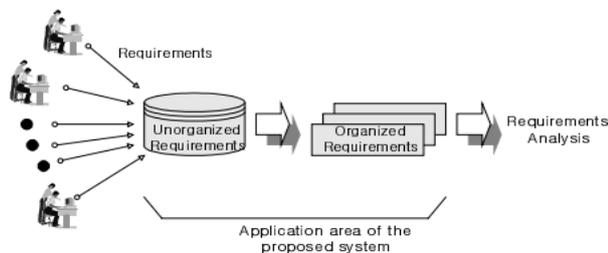


Fig. 2: Example of Unfinished and Finished Artifact [39]

Unfinished artefacts comprise of different types of artifacts in different formats containing inputs about requirements, represents voice of customers and other stakeholders and are captured during requirement development or management activity i.e. interview results, decision making sessions, policy, standard, changes, opinions, conflicts etc. [16, 33, 30, 35]. They are not organized instead their instances represent subset of requirements or its input. They represent repository of different types and formats i.e. voice, video, text, email etc and assist analyst in producing finished artefacts [?]. Various types of finished and unfinished artefacts are associated with each other in different capacity as well as Stakeholders are involved with these artefacts by various authorities [30].

## 1.2. Motivation

According to definition of Pre-RST, in order to describe life of requirements, all target artifacts representing finished artefacts are associated with other finished and unfinished artifacts representing source and intermediate artifacts. In order to follow this life these artifacts should be available in easy and navigating format. In this context if these contents are represented in some such a pictorial notations that can make these concepts more prominent and pop out the essence, analyst could achieve much benefit out of it.

On-going refinements can be shown by evolving and iterative representation. Different requirement development activities such as applying elicitation techniques i.e. conduct interview or workshops, and performing change activities, are visually associated with these refinements. The visualization structure can show association of any activity with any iteration, for example, first iteration of SRS is created after interview session which is conducted, question and answer session is recorded and then SRS is updated with new requirements and 2nd iteration of SRS is updated [2, 18, 30, 35]. All requirements traces from or towards origin can be represented by association lines and if cluster of links pointing to an origin can be shown by thick line. The visualized structure may

represent all aspects of any specific organized requirement life before its inclusion in a particular artefact and picture in graphical format. This visual model depicts easy comprehension of all aspect of requirement life. Similarly various types of source, intermediate and target artifacts can be represented by different metaphor or colours [2, 18, 20, 28].

The traceable information comprise of various types of artifacts, association nature between artifacts, activity types in which these associations establishes, people with different authorities involved, life of requirements, creation date of artefacts & activity, cluster of links associated with requirements, requirements associated with conflicts, with many contributors and requirements that are isolated. In order to achieve benefits of Pre-RST, analysts are required to capture traceability information discussed in previous paragraphs during the requirement process and certain visualization tools may renovate and model it using an improve visualized structure that may provide easy navigation and good comprehension. All above concepts should be represented by different visualized metaphor depending on which metaphor involves more perceptible inputs as well as consumes less cognition effort [2, 18].

Large cluster of textual representation of traceability information does not give good comprehension. It is difficult to comprehend textual or tabular formats and extracts useful information from it. However representing this traceability information in different visualized metaphor may realize many benefits and insights [2, 18, 20, 21, 28]. This paper will address several questions. Which traceability benefits can be achieved? How traceable information is useful in achieving these benefits? What visualized metaphors can pops out more prominent concepts and activate perceptual inputs? This section highlights the benefits of traceability and provides overview of traceability information required to capture for this purpose and can be mapped to some visualized model with prominent notations. Next section highlights the application of visualization in area of requirement traceability.

## 1.3. Visualization in requirement traceability

Visualization is defined as "the act of forming a mental vision, image, or picture of something not visible or present to the sight, or of an abstraction; to make visible to the mind or imagination" [1]. Gotel and Orlena [2] added that visualization is form of computing that renovates the different format of data in pictorial form for better perception, interpretation and insight of information. Visualization converts complex data in perceptual representation in order to boost our understanding and decisions. This figure out the stimulation of the sensory inputs of human. This process comprise of several steps. Initial step retrieves and extract the data and convert into some useful information; further steps map this information into graphical notations representing the pre-attentive attributes and then these attributes dynamically assembles in form of visualized model showing information in significant manner [2, 18, 19, 20, 21, 23, 27].

Visualization has major application in field of software engineering [2, 3, 24, 26, 27]. Various commercial tools are built generally as well as specifically for automating different life cycle activities i.e. project management, testing, debugging, modelling, requirement activities etc. These tools have capability to map these activities into different graphical notation and then converting them in visualized structure for quick interpretation [47, 30, 31].

Researchers [3, 18, 20, 21, 28, 1-2] emphasized on use of visualization techniques to address many issues in the area field of requirements engineering. These visualized models give better insight to different stakeholders in order to assist in decisions and findings. Gotel and Orlena [2] pointed out various benefits of visualization in requirement engineering activities such as requirement gathering, negotiations, managing quality in requirement artifacts and maintaining traceability. They described various graphical metaphor and notations to show different activities of requirements. UML provides many useful modelling notations in order to visualize requirement artifacts that may help in consensus

development between stakeholders and foster further design and development activities.

Specifically in area of requirement traceability, visualization process support lot of activities. Life of requirement is easily tracked from requirements artifacts towards its origin. Analyst can follow the depth of requirements. He can see the dependencies between different types of requirement artifacts. There will be better negotiation process between analyst and users when they can visualize production cycle of requirements as well as evolving nature of artifacts. Different benefits referred in literature are related to understanding, re-use, tracking, validation [2, 16, 18, 33] and conforming requirements on quality measures [2, 33, 30],

In this section we highlighted the usefulness of visualization and its application in area of Pre-RST. This section concludes the problems of Pre-RST and advantages of visualization for addressing Pre-RST issues.

#### 1.4. Significance of research

This paper represents Pre-RST traceability information with various graphical elements and translates. This will improve the realization of benefits for analyst and motivate them to capture traceability information during life cycle processes. Thus maintenance of Pre-RST becomes possible. Pre-RST has applications in various type of software projects and it is critically important for specific types of software domains e.g. safety critical system, aerospace system, business-critical system, regulatory system and other domain etc. and visualization process facilitates in achieving its benefits in these critical area. This research is evaluated by empirical evidence.

#### 1.5. Rest of sections

Rest of section of this document includes the research question & methodology, related work & discussion, review of literature, empirical evidence, conclusion, future direction & references.

**Research Question & Methodology:** This section discusses three research questions raised in this paper. Literature is reviewed to address these questions and results are evaluated using experiment and survey.

**Related Work & Discussion:** This section highlights the related state of art work on different techniques for achieving benefits of Pre-RST, and highlighted the research gap which justifies the research question.

**Literature Review:** This section highlights the results of literature reviewed for addressing three question and results are shown in various tables and figures.

**Empirical Evidence:** For achieving empirical evidence, an experiment is conducted on Project of Patient Registration of Dental Cloud to record traceability information in textual representation in tabular form. A Pictorial structure showing visualization of Pre-RST is proposed to represent the traceability information captured in textual representation, using the proposed visualized elements. To validate benefits of Pre-RST a survey is conducted, in which 10 analysts participated having various experience level and working on different nature of software projects, in order to compare the tabular textual structure and visualize model.

**Conclusion & Future Direction:** This section concludes the result obtained from literature review and empirically evidences and future work is discussed

## 2. Research questions and methodology

Methodology chosen for research broadly comprise of two parts. First part comprise of review of literature whereas 2nd part comprise of evaluation of results. First question is about to find most common benefits referred in literature. Second question shows list of traceability attributes required to capture in order to achieve traceability benefits. Third question is about visualized elements to represents the Pre-RST information that addresses the benefits

of traceability. For achieving empirical evidence, an experiment is conducted on Project of Patient Registration of Dental Cloud to record traceability information in textual representation in tabular form. A Pictorial structure showing visualization of Pre-RST, is proposed to represent the traceability information captured in textual representation, using the proposed visualized elements. To validate benefits of Pre-RST a survey is conducted, in which 10 analysts participated having various experience level and working on different nature of software projects, in order to compare the tabular textual structure and visualize model.

- Question 1: What benefits can be achieved from capturing traceability information during requirement development and management process?
- Question 2: Why traceability information in textual format is exhibit difficulty in achieving traceability benefits?
- Question 3: How visualization metaphor is useful in representing benefits of requirement traceability information?
- Question 4: What will be the usefulness of Visualization Structure to achieve traceability benefits for Pre-RST?

## 3. Related work and discussion

Lot of efforts are contributed by valuable researchers for almost more than two decades [2, 4, 14, 15, 16, 17, 18, 20, 46]. Research on Traceability is now going on diverse dimensions and many new chandleries are highlighted. Most of the researchers are contributing in the area of Post-RST [2, 3, 14, 15, 16, 21, 34, 40]. However Pre-RST has wide applications in life cycle processes [46]. Many problems of traceability arise due to difficulties in maintaining Pre-RS traceability [1, 4, 14, 16, 17, 24, 25, 26, 45, 34]. Various challenges relevant to Pre-RST have been reported in last two decades [1, 4, 6, 12, 16, 17, 21, 24, 26, 28, 29, 31, 35, 36, 45]. A recent survey is conducted on challenges related to Pre-RST.

Researchers have contributed in different facets, in earlier decade, contribution structure was proposed that defined different relationships between different contributing factor i.e. analysts and requirements. This contribution structure related different artefacts with their sources i.e. change requests and with contributors i.e. stakeholders [17]. At same decade few proposed Pre-RS techniques i.e. Pro-Art [14], Traceability Development Contract [11], Pre-RST Framework [?] and in recent decade, few proposed Concept Map Technique [?], Through Capability Engineering [?] etc. In recent years much of research is diverted towards developing of tools and technology as a solution to maintain Pre-RS traceability [?]. Many Wiki-based tools are getting popular [?, 30, 35] for hypertext-based linking sources. Further it is extended to hypermedia-based technology that offers linkages to various types of text and other media such as voice, video etc. [?].

One of the major problem reported [?] is related to lack of tangible benefits of traceability to analyst. Analyst is not much convinced to perform lot of manual action to store traceability information, as he is not well aware on benefits of storing information for traceability. To handle this problem, Traceability Contract Development was a proposed [11]. This technique highlights integration of traceability information in in development process. While integrating it with development process, analyst actions can be captured to automatically preserve the requirement artifacts. For Pre-RST it requires to integrate it with requirement engineering process models and techniques. If it is relied on analysts to record traceability information, since he is not convinced on its benefit, so he will ignore it because of tiring manual effort. Therefore it is a need is to reduce this manual effort whereas there should be some motivation factors also.

Gotel and Finkelstein [16] conducted a comprehensive survey on problems of requirement traceability, their benefits and issues involved in achieving it, and hundred software practitioners involved with experience of up to 30 years. Authors reported the multi dimension concerns of problems of traceability. They concluded that Pre-RST is more problematic due to many reasons.

They recommended improvements mainly in four areas; first was to increase awareness of traceable information; 2nd to create new abilities to obtain and capture traceability information; third was related to organizing and structuring of traceability information i.e. using hypertext editors etc.; fourth was to create presentation and visualization of traceability information i.e. by employing HCI factors and using animation for example links, textual or image etc. Comprehensive definition given by authors highlighted traceability benefits. This was a focus to realize analyst about benefits and use of traceability as a solution to addressing problems of Pre-RST. However this paper is not emphasizing about how to achieve these benefits.

Mäder, Patrick et al. [42] mainly focused on benefits of requirement traceability and empirically evaluated its usefulness on Post-RST. Authors performed a controlled experiment with seventy one practitioners involved in software maintenance activities while half of them are only maintaining requirement traceability process. They found that practitioners who were maintaining requirement traceability tasks are more fast and accurate than others. Authors concluded that it was saving of effort and improvement of quality of software maintenance. They reported few other benefits i.e. easier program comprehension and support for software maintenance when software changes. Authors reported that after many research findings that Pre-RST is more beneficial as and suggested that its benefits can easily be realized using some visualized notations. However no work is reported for achievement of Pre-RST benefits.

Ramesh and Jarke [33] proposed a meta-model of requirement traceability representing various types of requirement objects and four different types of association between them. Authors conducted a survey on 26 software firms to obtain empirical evidence. They differentiated between the users of requirement traceability on basis of their use and different benefits to them. They classified them in low and high end users while high end user comparatively require more dependencies, depth and detail path to trace requirements. Authors also represented few requirement components and association using some graphical notations. However a few benefits along with limited icons for representing requirement traceability were discussed.

Bouillon et al. [46] collected 29 regularly cited usage scenarios of requirements traceability and reported in a survey on both Pre-RST and Post-RST that 56 participants of our study that actually applied requirements traceability in practice use 42% of the 29 scenarios regularly. The results showed that mostly users are convinced for using requirement traceability. They found that the most common usages for requirements traceability are related with requirement development and management i.e. finding origin, locating rationale, and documenting history of requirement whereas related to Post-RST, tracking implementation state and analyzing coverage of requirement in source code, are also important. These four scenarios are used in around 90% of all projects either regularly or occasionally. They empirically evaluated the use and benefits of traceability and concluded that yet many benefits of traceability are not realized because of less tool usage. However role of visualization in achieving benefits of traceability is not discussed in this context.

Cleland et al. [15] proposed heterogeneous techniques to maximize the benefits of traceability and proposed TraCS (Traceability for Complex Systems) solution, a prototype tool that implemented this approach in order to optimize returns of the traceability effort and minimize the risk inherent to software evolution. In order to achieve objective and benefits of traceability, authors focused on some technological solution to capture Pre-RST at project level that could establish correct association between artifacts. Further concluded that role of automation can be useful in achieving correctness and coverage in requirement traceability. Role of visualization is not highlighted in this paper.

Rempel, Patrick, et al. [48] highlighted benefits of traceability process for developing safety critical and regulatory system. Authors identified the gap between various prescriptions of traceability guidelines and traceability process in practice. They further

highlighted ten types of traceability problems that may occur in projects and evaluated traceability approach on seven safety critical systems. It was found that none of the projects were conforming to traceability guidelines [48]. Graphical icons were used to explain the guidelines.

Decker et al. [30] highlighted the strengths of requirement wikis for problems of requirement traceability by providing navigational presentation of traceable information. Authors evaluated & compared wiki-based tools with different types of traceability tools and concluded that wiki-based tools provide easy navigation to access and maintain traceability information i.e. easy access to previous artifact history, integration of different stakeholder views for example tagging conflicts, decisions etc. Many other benefits of wikis are mentioned in this paper. Analyst can easily add any association and tag any contribution like conflict or his opinion. Minimum or no association indicates unfocused discussion; either artifact is not linked with each other. It also says that artifact is either unreliable.

Other researchers also indicated the benefits of wikis [35]. Wiki is effective tool to organize traceability information however it is much complex to follow all these textual links. Many benefits cannot be realized from these textual links. If an attempt is made to show all associations maintained by wiki tools, it would show a complex visualization with cluster of association. However Gotel and Finkelstein [16] noted that organization and visualization are two dimensions.

Cleland-Huang et al. [3] emphasized on benefits of the various visualization techniques for validation of candidate traceability links between requirement artifacts, created by information retrieval techniques. These techniques are used to automate traceability process to establish traceable links between candidate artifacts which grows frequently and dynamically and hence large cluster of traceable links are established. Authors pointed the benefits of visualizations as it improves the speed and accuracy and capable analysts to validate candidate traceability links. The focus on this paper on Post-RST and only association lines are as a metaphor here for representing traceability information. However research proposes lot of perceptual elements can assist in achieving benefits of traceability.

Gotel and Orlena [2] emphasized on use of visualization techniques and discussed the strengths of various representation metaphors to manage requirement problems. According to authors UML provides many useful modeling notations in order to visualize requirement artifacts that may help in consensus development between stakeholders and foster further design and development activities. They recommended different graphical metaphors for this purpose. However their focus is on broad problems of requirement management and authors only focused on benefits of visualization for traceability and specifically didn't propose any visualization elements for Pre-RST.

Marcus et al. [20] highlighted various advantages of visualization in achieving benefits of requirement traceability. They proposed different types of colours and shapes; however Pre-RST is not focused. Merten et al. [18] evaluated various visualization techniques and proposed sunburst and net-map visualization to represent various concepts of Pre-RST. They demonstrated benefits of visualization using these techniques and highlighted future research direction to map various benefits of traceability with different visualized elements discussed in literature. A big research gap is highlighted here that may improve navigation and presentation of Pre-RST using various visualization techniques.

### 3.1. Research gap

After a comprehensive review of related work, we can categorize our study in four directions. In first direction lot of papers are highlighted benefits of requirement traceability and noted various problems on the way, defined meta-model for it and proposed different techniques for capturing traceability information. In second direction, to overcome manual tasks and to motivate the analyst for benefits of traceability, they proposed various information

retrieval techniques that provide automation in capturing traceability information. 3rd direction of work introduced various wiki-based tool to organize the complex information of traceability. In fourth direction, various visualized models are proposed to overcome the problems of traceability and to realize the benefits by visualization elements.

Work in all directions is progressing to achieve benefit of traceability. A research gap found here is related to motivate of analyst in achieving benefits of traceability using visualized models. Much work is done in area of Post-RST and many visualized models are proposed here, however yet representation is limited to few visualized elements. Techniques for automatic links retrieval creates tangle of links and it become difficult to manage either to edit or remove irrelevant links manually. Our paper proposed visualized structure to address these issues.

However progress in area of Pre-RST is limited. No such visualized model is proposed specifically to address Pre-RST problems. Many benefits of Pre-RST can achieved by visualized models that may represents various traceability information in such a visualized metaphor that can assist analyst to accessing all the traceability information in easy and quick manner.

### 3.2. Discussion: Benefits and application

Pre-RST has application in various domain and they conform various standards. For development of safety-critical systems, other regulatory system, business critical system, aerospace industry, railway communication, signaling, medical, automotive and processing systems etc. A certified development process is required that should conform on process standards like CMMI, SPICE, IEEE, SEI [22, 42, 43, 44, 46, 15, 48]

Lot of benefits highlighted in definition of traceability as well highlighted in literature can be achieved by analyst by follow the life of requirements during its production to identify various artifacts, their types, their nature, time of creation, nature of activities, evolution history of requirements and its refinement cycles and contributors involved in production cycle of requirements.

Figure 3 is demonstrating activity of meeting cycle by few graphical metaphors. Association lines are showing order of artefact creation. Two types of rectangle boxes are shown here. Chronological order is representing evolution process which that shows that artefact is refined number of time. Multiplicity of instance shows the number of time artefact refined. Three meetings are initiated. Once instance is shows one refinement. Arrow is representing completion of activity. Many association lines are showing confidence on final artifact since it shows lot of refinements after many stakeholders contributed. Now analyst can easily interact with any activity or artefacts in quick manner. He can focus on the required information by zooming and filter rationale of any activity [?]

Further Pre-RST is also useful to trace and follow the use cycle of requirement. It is useful for post development phase of requirement where requirement is used for design, code, test, deployment and maintenance. Post-RST is useful for tracing the design, code, test, deployment and maintenance progress of requirements in forward direction. It is also useful to trace requirements for their understandability, clarity, changeability, completeness etc. from design, code, test and maintenance in backward direction. Many other benefits like assurance, reliability, and verifiability of requirements during other life cycle processes or during or after software use or re-usability for any other project etc. can be achieved if Pre-RST is maintained, so role of Pre-RST is important for both requirement production and use [42].

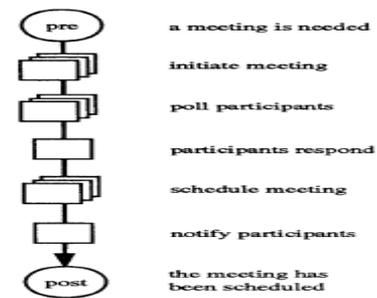


Fig. 3: life cycle of artefact

## 4. Literature review

This section address the research question and show the detail results after a comprehensive review.

### 4.1. Question No. 1: What benefits can be achieved from capturing traceability information during requirement development and management process?

Literature discussed various benefits of Pre-RST [16, 18?]. Most of researchers highlighted it as a motivational factor [11, 16?] whereas some discussed requirement wikis to access and navigate requirements and their sources to achieve benefits of traceability. Some also highlighted and reported empirical evidence of the usefulness of information retrieval techniques to automate requirement traceability process [3,?]. Few researchers discussed the gain of benefits through visualized representation of traceability information of Post-RST [3,?]. However few such graphical elements are discussed or used in visualization for representation. Literature reported various types of pre-attentive attributes that provide such a visualized model that may stimulate human perceptual elements for comprehension instead consuming cognitive effort such as line, width, size, colors, motion etc [?]. It is not required to pay attention when an object is moving. Similarly it is not mental stress is required to differentiate between different shapes. Various types of such elements are reported in research [2, 3, 18, 20, ?].

In this section we extracted and compiled various traceability benefits that are source of motivation factors for analyst [2, 3, 15, 16, 18, 20, 30, 33, 42, 43, 46, 47]. see table 1. In order to demonstrate the benefit by visualized representation, we filtered this list and selected nine benefits for purpose of sampling and experimentation, see table 2. In next section we identified traceable information that is required to be represented in order to achieve the selected benefits.

Objective of this research activity is to demonstrate the benefits of traceability using some visualized model. Further goal is make the process of Pre-RST beneficial and easily to put in practice. The list of benefits extracted and compiled from various sources is that shown in table 1.

Table 1: Benefits of requirement traceability

No.	List of traceability benefits
1	to visible the changes in finished artifacts
2	to track those who are involved in changes
3	to trace artifacts of those team members who is not present either left the organization or on leave
4	any substitute person can trace in depth
5	managing conflict resolution in presence of traces of artifacts i.e. trace and re-open artifacts for decision making or some negotiation and in presence or absence of concerns
6	help to identify or locate the essential participants
7	show good collaboration of work i.e. effective or weak
8	re-usability of artifacts and knowledge 8) different use to different users on demand
9	required details of requirements
10	life of requirements
11	detail history of refinements from version history
12	maturity of version control process
13	maturity of artifacts

14	audit of practitioners & analysts i.e. how work done in specific time period, who is responsible for what
15	improved time to and correctness of solution
16	easier program comprehension
17	assurance of maturity i.e. meeting design constraints, rationale
18	support of manage changes during requirement process 19) support of manage changes during life cycle process i.e. code, design i.e. when changes occur at any time
19	performance of practitioners i.e. faster and creating more correct solutions after visualization
20	analysts feel supported by traceability in performing their tasks i.e. often responding very enthusiastic towards traceability.
21	traceability supports the certification process for some domain i.e. safety critical domain etc.
22	mandated by several standards
23	follow depth of requirement i.e. to know all sources or origins
24	follow breath of requirement i.e. to know all related requirements, FR, NFR i.e. other types security, safety , performance etc.
25	visualize coverage of traceability i.e. correct, precision
26	facilitate maintenance of traces generated automatically i.e. edit or delete irrelevant, redundant or incorrect path, missing artifact creation activity, missing trace path, ambiguous trace path, contradicting artifact creation after merge, ambiguous trace path after merge, missing artifact type, missing trace path, invalid trace path, ambiguous trace path, missing trace link
27	demonstration of rigorous process i.e. provide evidence that system is safe and secure for use
28	to trace the information that is not traceable
29	information inside the artifact is also traceable like requirements, performance, goals etc.
30	activity produces artifact can also be traced i.e. artifact is origin or destination
31	follow if conforming technical guidance
32	Assessment from different links types.
33	follow to see dependency between artifacts i.e. who derive what
34	relevance and usage intensity of the scenario
35	finding compliance of regulatory standards
36	Confidence on specific requirement for use and re-use
37	Conforming to quality attributes
38	Reliability of requirements

Literature highlighted lot of constraints in achieving traceability benefits i.e. lack of motivation, imbalance between cost and time, time consumed for capturing traceable information, habits of not capturing information such as defining requirement and traceable information [?]. Capturing traceability information is major constraints among all others. Many issues are related to organizing, navigations, filtering etc. However we are focusing on addressing all these issues and achieving benefits by mapping and representing traceable information by using visualized metaphor.

**Table 2:** various benefits of traceability and how they can be achieved

No	Traceability benefits	Descriptive achievement of traceability benefits
1	Follow Life of Requirement	Requirement described in finished artefact is associated with activity i.e. elicitation techniques in which is captured and source and intermediate artifacts from where it is captured. People are associated with activities and sources as a contributing factors
2	Confidence on Requirements	If life of requirement can be followed then user have full confidence on re-using it or producing it .
3	Reusability of Requirements	To re-use requirement of earlier software project, analyst can trace it to sources and read it and evaluate its authenticity
4	Audit of Analyst Performance	Analyst performance can be tracked from number of sources he used to write specific requirements and number of techniques he performed
5	Reliability of Requirements	Those requirements are reliable on which analyst is confident and there is no much gap between creation date of source and target artifacts
6	Verifiability of Requirements	To verify requirements, its input artifacts should be available
7	Validation of	To validate the finished artifacts, its source or

	Requirements	intermediate artifacts and its contributors roles should be available
8	Maintenance	To add new, modify existing or remove irrelevant traceability links
9	Conforming to Quality Attributes	Traceability information can also ensure conformity of requirement on quality attributes if semantics is applied
		Complete - If requirement is not linked with any source or activities, then perhaps analyst have to add some information in order to complete it - If any stakeholder tags any conflicts with the requirement, then we cannot say that requirement is correct. Because stakeholder may willing to share some inputs
		Understandable etc. Requirements are understandable by using various types of sources associated with it.

Figure 2 shows comprehensive list of traceability benefits. Many benefits are yet not chosen for sample, see figure 2 and future research can extend this experimentation further. Many benefits can derive from other benefits i.e. benefits at no. 36 noted in table 1 can be achieved if benefits at 33, 32, and 2 are achieved. This dependency relation is not covered in this paper. It will be a future research direction. In this paper our focus is to illustrate the benefits of Pre-RST using visualization, therefore we selected few benefits that can be achieved by visualization. Our claim that all almost all benefits can be realized by visualization will be evaluated in future work.

**4.2. Question 2: Why traceability information in textual format is exhibit difficulty in achieving traceability benefits?**

As discussed in previous section that in order to achieve benefits using visualization, it is required to represent and map traceability information using some visualized notations. Our focus of discussion in this paper is not about efficiency of traceability techniques, we have to extract and compile the list of traceability attributes so that it can be mapped to traceability benefits, further we can represent these traceability attributes in both textual and visualization format. As we discussed earlier that it is very time consuming process for analyst when he navigates and filters required information from various artifacts and their sources. Because he has to deal with cluster of associations linking these artifacts [3] and this makes his motivation low and thus it make difficult to release benefits from traceability process.

**Table 3:** Types of traceability attributes

No	Traceability Attributes with Example
1	category of artifacts i.e. source artifacts, intermediate artifacts, destination artifacts, target artifacts, origin artifact
2	types of requirement artifacts i.e. SRS, Vision, Business Case, Use Case, Requirement Rationales, Problem Statements, user stories, goals
3	association links between source and target artifacts
4	different format of artifacts i.e. text, images, videos, email etc.
5	organization of artifacts i.e. finished and unfinished
6	requirement activities performed i.e. elicitation, analysis, negotiation, change management techniques, manage change, manage resolution and conflict
7	types of operation/activity performed on artifacts i.e. create new, add or edit requirements etc.
8	requirement knowledge is organized in different sections in finished artefacts i.e. instance of requirement e.g. FR 1.1, OP 2.2
9	different versions i.e. version 1, 2, 3
10	date of operations i.e. Jan 01, 2016, Feb 02, 2016 etc
11	types of associations i.e. dependency either derived or served as input to other finished artefacts
12	stakeholder types i.e. contributors

13	create or use of artifacts
14	stakeholder roles i.e.
15	flow of information
16	relation between different finished artifacts
17	size of artifacts
18	Name of participants
19	Operation performed on size of files
20	Name of artifacts
21	Artifact will either user only or created
22	Location of artifact i.e. location in drive etc

In order to strength our claim we have conducted a experiment in this paper. In this section, a detail literature survey is conducted and comprehensive list of traceability attributes are extracted [?] see table 3. However we choosing some attributes as sample for purpose of further experimentation. This experiment is conducted on software project of dental cloud and using these traceability attributes defined in table 4, traceability information is captured and represented in textual tabular format, see next section. Results are noted in table 6.

Different tools are available that can automatically capture the traceability information, however no mature tools is in practice. Since our purpose is only experimentation, therefore we conducted a study and executed a software project to capture traceability information against these attributes. We maintained these information in table, see table 6. Because our objective is to compare and evaluate the textual representation with visualized representation in order to demonstrate the traceability benefits. We assumed that any tool can be used for such purpose, however in future work, new tool can be proposed that may capture traceability information during development process.

**Table 4:** Types of traceability attributes selected as a sample for experiment to generate data set

No	Chosen Traceability Attributes	Description
1	Target Artifact Type Name	The artefact created as target artefact after using the source inputs. This artefact can be both finished or unfinished format. The artifacts in progress will named as target artefact. This artifact can also served as intermediate or source artefact for any other artifacts.
2	Artifact Type	SRS, Feature List, Goals, User Story etc.
3	Organized	Artifacts are created and maintained under version control are finished artifacts, Unfinished artifacts are not maintained by version control process, it will remain one instance. It will be created by team or fetched only for use.
4	Version No	Finished artifacts evolves in various versions
5	Action Performed on Requirement Instance No i.e. FR 1.1, 1.2	Certain action can be performed such to add any new requirements, to edit, or change or to delete requirement.
6	Source Artifact Name	Any finished or unfinished
7	Activity Conducted / Relation	Interview, Brainstorming, tagging any opinion
8	Activity Purpose	To negotiate, prioritize, change, report conflict
9	File Size	Show the size of file e.g. 1.2 MB
10	Creation Date	When this artefact is operated either created, or modified i.e. 01/01/2015
11	Contributor (s)	Person who directly involved
12	Contributor (s) Roles	Job Role i.e. Business Analyst

#### 4.3. Question 3: How visualization metaphor is useful in representing benefits of requirement traceability information? What will be the usefulness of Visualization Structure to achieve traceability benefits for Pre-RST?

In order to study the usefulness of various metaphor reported and experimented, we conducted a detail literature survey and identified different graphical notations and metaphor that are described in literature for different use and application [?]. We already shared these details in previous sections. We selected some graphical icons and metaphor on basis of their usefulness in our domain. After a detail literature survey [?], from list of traceability benefits noted in table 2, and traceability information described in table 4, various pre-attentive elements in different graphical notations are evaluate for their usefulness in our domain. Table 5 shows list of various metaphor that can be used to represent and visualize traceability information in order to assist analyst in achieving traceability benefits.

Next section demonstrates empirical evidences collected after experimentation and survey gained. From execution of project we created we textual representation of traceability information in table 6. We used this textual representation and mapped it to list of graphical metaphor and icons described in table 5. We finally proposed a visualized structure that is mapping textual information into such graphical icons that are representing pre-attentive and look and feel elements, see figure 5. In order to validate, this representation of a visualized model is further compared with textual representation on basis of a survey conducted. Further results are noted in conclusion section. This visualized model also provides interactive use of traceability information.

## 5. Empirical evidence

### 5.1. Question 4: What will be the usefulness of Visualization Structure to achieve traceability benefits for Pre-RST?

Experimentation on Dental Project to record Pre-RST information in tabular form: A software project on dental cloud patient registration is executed for a purpose to capture traceability information against the traceability attributes described in table 4, during requirement development and management process. The objective of this experiment is to obtain textual data of traceability information. This table is also used as a data set for comparison with proposed visualized model, see figure 6. In first iteration Demo.mp3 is download from some source i.e. google etc. (this file/artifact is not create and no source is recorded because this file in not created by our team, however this file is only for use purpose). Other information is captured against traceability attributes (see table 4) are artifact type i.e. video, organized i.e. unfinished because it will not evolve and it will not kept under version control, no action is performed on it else it is archived and used, no its source is recorded, activity conducted on this file is the only browse and archive, purpose of this activity is to gain knowledge from, size of this artifact is 3.5 MB, creation date of this file is aug 01, 2010, location address is noted and contributor is person who make it available i.e. Sohaib.

Demo.mp3 is used as source for questionnaire.doc, this file/artifact is archived and it source artifact is demo.mp3. Other information is captured against traceability attributes (see table 4) are artifact type i.e. text, organized i.e. unfinished because it will not evolve and it will not kept under version control, some action is performed on it i.e. it is newly created and archived and used, its source is recorded i.e. demo3.mp3, relationship with demo.mp3 is i.e. analyst read the demo.mp3 to create questionnaire.doc. purpose of this activity is create questions, size of this artifact is 1.1 MB, creation date of this file is dec 30, 2015, location address is noted and contributor name is Ahmed, role is sales person.

Questionnaire.doc is used as source for InterviewSession.doc, this file/artifact is archived and it source artifact is Questionnaire.doc. Other information is captured against traceability attributes (see table 4) are artifact type i.e. text, organized i.e. unfinished because it will not evolve and it will not kept under version control, some action is performed on it i.e. it is newly created and archived and used, its source is recorded i.e. Questionnaire.doc, relationship with Questionnaire.doc is i.e. analyst compiled this interview form

Questionnaire.doc, purpose of this activity is to complete results, size of this artifact is 0.2 MB, creation date of this file is jan 01, 2015, location address is noted and contributors name is Sohaib and Ahmed, role lead analyst and sales person.

InterviewSession.doc is used as source for SRS\_1.doc, this file/artifact is archived and its source artifact is InterviewSession.doc. Other information is captured against traceability attributes (see table 4) are artifact type i.e. text, organized i.e. finished because it will evolve and it will be kept under version control, some action is performed on it i.e. it is newly created and archived and used, and four functional requirements are created i.e. FR 1.1 to 1.4, relationship with InterviewSession.doc is i.e. analyst compiled from interview results, purpose of this activity is to study the registration form fields, size of this artifact is 0.3 MB, creation date of this file is jan 03, 2015, location address is noted and contributors name is Sohaib, role is lead analyst.

Now reading above paragraph and extracting traceability data is much difficult. Tabular representation shown in table 6 is also much difficult to read in order to achieve any benefit. Even if we represent it in more simple form, no big change in result will occur. In next para, a visualized diagram is explained.

Visualized Structure representing traceability benefits

A visualized structure showing a detail representation of traceability information extracted from table 6. This visualized structure is showing different types of artifacts with different icons i.e. icon for SRS, icon for video etc. The nature of artifacts with different colors i.e. finished artifacts are shown with white and unfinished are shown with dark color. If artifact is maintained under version control is shown in chronological order. Long arrow is showing that artifact is created since long time. Thick arrow shows density of sources. Labels are used to manage textual information only when user focus any object. Label shows name of artifact, its creation date and its iteration number. In order to manage cluster of arrows, arrows associated with same type activity i.e. interview techniques, are bundled. We created textual representation of traceability information in table 6. We used this textual representation and mapped it to list of graphical metaphor and icons described in table 5. Research Question & Methodology: This section discusses three research questions raised in this paper. Literature is reviewed to address these questions and results are evaluated using experiment and survey.

Lot of other traceability benefits can also be demonstrated from this visualization structure and a survey is conducted to validate this research findings. To evaluate the usefulness of visualized metaphor, this section concluded result of achievement of traceability benefits of visualized structure. This section will describe the empirical evidence obtained. This step is performed in two phases. First step identifies the different graphical notations and metaphor that can be used to visualize traceable information in order to assist analyst in achieving traceability information, see table 5. In 2nd step, we will use textual representation of traceability information in table 6 and using list of graphical metaphor and icons, we created a visualized structure that is mapping textual information to graphical icons, representing such a visualize model that provide all benefits providing look and feel sensation. It may also provide interactive use of traceability information. Further a survey is conducted to validate the results.

In order to strengthen our claim we have conducted an experiment in this paper. In first phase we conducted a detail literature survey and comprehensive list is obtained see table 3. However we will use only sample for our experiment. To justify the complexity level, in second phase we executed a project while recording traceability attributes define in table 4 and recorded the traceability information in textual representation in tabular format, see table 6.

To evaluate the usefulness of visualized metaphor, this section shows achievement of traceability benefits of visualized structure. This section will describe the empirical evidence obtained. This step is performed in two phases. First step identifies the different graphical notations and metaphor that can be used to visualize

traceable information in order to assist analyst in achieving traceability information, see table 5.

Visualized Structure representing traceability benefits

Lot of other traceability benefits can also be demonstrated from this visualization structure. However in order to validate our research findings. A survey is conducted from ten requirement analyst having experiences from one year to five years, working on different nature of software project. They compared both textual representation in tabular form and visualization structure and result is evaluated on basis of four other questions that are 1) their experience with maintaining Pre-RST information 2) what type of information they maintain during traceability 3) if they recognize any benefit from maintaining traceability 4) if they realize any benefit from visualization.

Result showed that only 7 out of 10 analysts are maintaining traceability but with few recording few information. Result also showed that they feel much time and effort consumption in maintaining traceability. They also highlighted imbalance between effort and benefit. However the few who are not using traceability are also on same page. In the last part of survey they compared both textual representation in tabular form (see table 6) and visualization (see figure 5). They concluded about textual representation that it is so complicated and hard to trace. When they are asked about the benefits shown in table 2, they replied that it is difficult to realize these benefits. However the visualization structure is much convinced. They have taken lot of interest in this diagram. More over they contributed many other issues that can be addressed from this visualization, however taken into account for future work.

## 6. Conclusion and future direction

In this paper we have proposed an improved visualization structure to represent Pre-RST information. The visualization realized various benefits of requirement traceability. It is effective demonstration to developers which convince them in future to capture traceability information during life cycle development processes. This paper highlighted various benefits that can be achieved from maintaining traceability information. Different traceability attributes are also compiled from research. This paper proposed various visualized metaphor that are representing benefits of Pre-RST. Work of many researchers in this context is also discussed.

In order to evaluate this visualization empirically, a software project is executed on dental cloud patient registration system, and Pre-RST information is captured and shown in tabular textual representation. This textual representation is further compared with visualized representation proposed during a survey conducted in this paper. The result was so encouraging that analyst even contributed many of their real life problems to be addressed with this visualization. They showed lot of motivation for maintaining traceability information.

This paper also highlights many future directions. This experimentation should be extended to different nature of software projects. Many other traceability information should also be integrated for such experimentation. This paper highlighted a big list of benefits. New visualization should be proposed in order to demonstrate these benefits. Scope of survey should be extended to experienced practitioners and with new questions.

## References

- [1] Oxford English Dictionary, 1989
- [2] Gotel, Orlena CZ, Francis T. Marchese, and Stephen J. Morris. "On requirements visualization." *Requirements Engineering Visualization*,
- [3] 2007. REV 2007. Second International Workshop on. IEEE, 2007. Cleland-Huang, Jane, and Rafal Habrat. "Visual support in automated tracing." *Requirements Engineering Visualization*, 2007. REV 2007. Second International Workshop on. IEEE, 2007.

- [4] Walia, Gursimran Singh, and Jeffrey C. Carver. "A systematic literature review to identify and classify software requirement errors." *Information and Software Technology* 2009. 1087-1109.
- [5] Martin, Sacha, et al. "Requirements engineering process models in practice." 7th Australian workshop on requirements engineering. Deakin University, Melbourne, Australia. 2002.
- [6] Faily, Shamal, et al. "Requirements sensemaking using concept maps." *Human-Centered Software Engineering*. Springer Berlin Heidelberg, 2012. 217-232.
- [7] Gotel, Orlena, and Anthony Finkelstein. "Extended requirements traceability: results of an industrial case study." *Requirements Engineering*, 1997
- [8] Pohl, Klaus. "PRO-ART: Enabling requirements pre-traceability." *Requirements Engineering*, 1996, Proceedings of the Second International Conference on. IEEE, 1996.
- [9] Chikh, Azeddine, and Mashael Aldayel. "Reengineering Requirements Specification Based on IEEE 830 Standard and Traceability." *New Perspectives in Information Systems and Technologies*, Volume 1. Springer International Publishing, 2014. 211-227.
- [10] Gough, Paul A., et al. "Scenarios-an industrial case study and hypermedia enhancements." *Requirements Engineering*, 1995. Proceedings of the Second IEEE International Symposium on. IEEE
- [11] Arkley, Paul, and Steve Riddle. "Overcoming the traceability benefit problem." *Requirements Engineering*, 2005. Proceedings. 13th IEEE International Conference on. IEEE, 2005.
- [12] Arkley, Paul, Paul Mason, and Steve Riddle. "Position paper: Enabling traceability." *Proceedings of the 1st International Workshop on Traceability in Emerging Forms of Software Engineering*, Edinburgh, Scotland (September 2002). 2002.
- [13] <https://codebeamer.com/>
- [14] Pohl, Klaus. "PRO-ART: Enabling requirements pre-traceability." *Requirements Engineering*, 1996, Proceedings of the Second International Conference on. IEEE, 1996
- [15] Cleland-Huang, Jane, Grant Zomont, and Wiktor Lukaszik. "A heterogeneous solution for improving the return on investment of requirements traceability." *Requirements Engineering Conference*, 2004. Proceedings. 12th IEEE International. IEEE, 2004.
- [16] Gotel, Orlena CZ, and Anthony CW Finkelstein. "An analysis of the requirements traceability problem." *Requirements Engineering*, 1994., Proceedings of the First International Conference on. IEEE, 1994.
- [17] Gotel, Orlena, and Anthony Finkelstein. "Contribution structures [Requirements artifacts]." *Requirements Engineering*, 1995., Proceedings of the Second IEEE International Symposium on. IEEE, 1995.
- [18] Merten, Thorsten, Daniela Jüppner, and Alexander Delater. "Improved representation of traceability links in requirements engineering knowledge using Sunburst and Netmap visualizations." *Managing Requirements Knowledge (MARK)*, 2011 Fourth International Workshop on. IEEE, 2011.
- [19] Ernst, Neil, Yijun Yu, and John Mylopoulos. "Visualizing non-functional requirements." *Requirements Engineering Visualization*, 2006. REV'06. First International Workshop on. IEEE, 2006.
- [20] Marcus, Andrian, Xinrong Xie, and Denys Poshyvanyk. "When and how to visualize traceability links?." *Proceedings of the 3rd international workshop on Traceability in emerging forms of software engineering*. ACM, 2005.
- [21] Cleland-Huang, Jane, et al. "Utilizing supporting evidence to improve dynamic requirements traceability." *Requirements Engineering*, 2005. Proceedings. 13th IEEE International Conference on. IEEE, 2005. [22] Aizenbud-Reshef, Neta, et al. "Model traceability." *IBM Systems Journal* 45.3 (2006): 515-526.
- [22] Pichler, Mario, and Hildegard Rumetshofer. "Business process-based requirements modeling and management." *Requirements Engineering Visualization*, 2006. REV'06. First International Workshop on. IEEE, 2006.
- [23] Schneider, Florian, Helmut Naughton, and Brian Berenbach. "A modeling language to support early lifecycle requirements modeling for systems engineering." *Procedia Computer Science* 8 (2012): 201-206.
- [24] Wnuk, Krzysztof, Björn Regnell, and Lena Karlsson. "What happened to our features? Visualization and understanding of scope change dynamics in a large-scale industrial setting." *Requirements Engineering Conference*, 2009. RE'09. 17th IEEE International. IEEE, 2009.
- [25] Herrmann, Korbinian. "Visualization of release planning." *Requirements Engineering Visualization*, 2006. REV'06. First International Workshop on. IEEE, 2006
- [26] Eppler, Martin J., and Ken W. Platts. "Visual strategizing: the systematic use of visualization in the strategic-planning process." *Long Range Planning* 42.1 (2009): 42-74.
- [27] Cleland-Huang, Jane. "Toward improved traceability of non-functional requirements." *Proceedings of the 3rd international workshop on Traceability in emerging forms of software engineering*. ACM, 2005. [29] Asuncion, Hazeline U., Arthur U. Asuncion, and Richard N. Taylor. "Software traceability with topic modeling." *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*. ACM, 2010.
- [28] Decker, Björn, et al. "Wiki-based stakeholder participation in requirements engineering." *Software*, IEEE 24.2 (2007): 28-35
- [29] DOORS, Web Page: Dynamic Object Oriented Requirements System, <http://www.telelogic.com/doors>.
- [30] [www.ibm.com/software/products/en/ratidoor](http://www.ibm.com/software/products/en/ratidoor)
- [31] IEEE Computer Society. *Software Engineering Standards Committee*, and IEEE-SA Standards Board. "IEEE Recommended Practice for Software Requirements Specifications." *Institute of Electrical and Electronics Engineers*, 1998.
- [32] Ramesh, Balasubramaniam, and Matthias Jarke. "Toward reference models for requirements traceability." *Software Engineering*, IEEE Transactions on 27.1 (2001): 58-93.
- [33] Nair, Sunil, Jose Luis de la Vara, and Sagar Sen. "A review of traceability research at the requirements engineering conference re@21." *Requirements Engineering Conference (RE)*, 2013 21st IEEE International. IEEE, 2013.
- [34] Geisser, Michael, et al. "New Applications for Wikis in Software Engineering." PRIMIUM. 2008.
- [35] [www.sei.cmu.edu](http://www.sei.cmu.edu)
- [36] *Process Traceability-Systematic Review*, Nitin, Vivekanand S Gogi, Dr. N S Narahari, *International Journal of Innovative Research in Science, Engineering and Technology*
- [37] *Center of Excellence for Software Traceability*, [www.coest.org](http://www.coest.org)
- [38] Ko, Youngjoong, Sooyong Park, and Jungyun Seo. "Web-based requirements elicitation supporting system using requirements categorization." *Proceedings of Twelfth International Conference on Software Engineering and Knowledge Engineering (SEKE 2000)*, Chicago, USA. 2000.
- [40] Walia, Gursimran Singh, and Jeffrey C. Carver. "A systematic Literature review to identify and classify software requirement errors." *Information and Software Technology* 51.7 (2009). 1087-1109.
- [41] <http://www.standishgroup.com/>
- [42] Mäder, Patrick, and Alexander Egyed. "Do developers benefit from requirements traceability when evolving and maintaining a software system?." *Empirical Software Engineering* (2014): 1-29.
- [43] ISO (2011) ISO:26262-6:2011 Road vehicles - functional safety - part 6: Product development at the software level
- [44] RTCA / EUROCAE. "Software Considerations in Airborne Systems and Equipment Certification", DO- 178C/ED-12C, (2011).
- [45] Ahmad, Arslan, and Muhammad Ahmad Ghazali. "Documenting requirements traceability information for small projects." *Multitopic Conference*, 2007. INMIC 2007. IEEE International. IEEE, 2007.
- [46] Bouillon, Elke, Patrick Mäder, and Ilka Philippow. "A survey on usage scenarios for requirements traceability in practice." *Requirements Engineering: Foundation for Software Quality*. Springer Berlin Heidelberg, 2013. 158-173.
- [47] Gotel, Orlena, and Patrick Mäder. "Acquiring tool support for traceability." *Software and Systems Traceability*. Springer London, 2012. 43-68.
- [48] Rempel, Patrick, et al. "Mind the gap: assessing the conformance of software traceability to relevant guidelines." *Proceedings of the 36th International Conference on Software Engineering*. ACM, 2014.