A TOOL FOR MANAGING SOFTWARE TRACEABILITY AND SUPPORTING TQL: TRACE QUERY LANGUAGE

Andy Cook, and Jane Cleland-Huang*, School of Computing, DePaul University, Chicago, IL, 60604, jhuang@cs.depaul.edu

Requirements traceability, defined by Gotel and Finkelstein as the "ability to follow the life of a requirement in both a backward and forward direction" is a critical element of any rigorous software development process. For example, the U.S. Food and Drug Administration (FDA) states that traceability analysis must be used to verify that the software design implements the specified software requirements, that all aspects of the design are traceable to software requirements, and that all code is linked to established specifications and test procedures. Similarly, the Federal Aviation Administration (FAA) has established DO-178B as the accepted means of certifying all new aviation software, and this standard specifies that at each and every stage of development "software developers must be able to demonstrate traceability of designs against requirements." Unfortunately, there is almost universal failure across both industry and government projects to implement successful traceability. These difficulties have been attributed to problems associated with creating, maintaining, and utilizing requirements traceability matrices, and also attributed to the perception by many developers that the effort of tracing exceeds the benefits it returns. Traceability plays an important role in the development and assurance of software systems that permeate our society. It helps to assure that an as-built system correctly implements all requirements by supporting change impact assessment, re-engineering of applications, and other critical software engineering activities. To address these problems researchers have introduced a powerful traceability framework, Traceability$^+$, that is designed to deliver value-added traceability services to support the software engineering tasks of a diverse set of stakeholders. Traceability$^+$ enables trace stakeholders to not only model and execute basic traceability links, but to execute more advanced queries to directly support tasks related to requirements validation, impact analysis and other core software engineering activities. One of the most important aspects of the Traceability$^+$ framework is its emphasis on modular design. Utilizing the existing NetBeans Platform and common open-source principles, Traceability$^+$ allows any institution the ability to quickly and efficiently create and implement a customized traceability tool.

In this abstract we introduce the initial component of the Traceability+ tool, which allows project stakeholders to build and manage traceability graphs that model the set of traceable artifacts and their core traceability links. This tool's interface mimics that of other popular graph editors with a familiar shape palette panel, birds-eye-view feature, auto-alignment, and font adjustment. With these features the user can create an accurate visual model of a trace matrix. This visual model can than be linked to actual trace data in the form of XML documents. In addition, the tool is designed with extensibility in mind, and future versions will provide an environment for modeling complex multi-link trace queries using a new Trace Query Language (TQL) built on top of XPath and XQuery.

The current version of the tool was evaluated through modeling the traceable components of a 30,000 requirement safety critical system.

Acknowledgments
This research project was funded by REU supplement grant # 0936417.