Complex software systems contain a large number of interconnected development artifacts such as requirements, design models and source code. Traceability enables understanding and managing these artifacts. Establishing traceability is not a trivial task, it requires the development company to plan how traceability fits into its processes and provide tools for traceability establishment. In practice, guidelines for establishing traceability are lacking, therefore companies struggle with establishing and making the most of traceability.

The objective of this research is to improve software traceability tools and processes. We conducted empirical studies to understand practitioners’ traceability challenges and propose solutions for four challenges: manual work of establishing traceability, lack of configurable tools, diverse artifacts and tools, and unclear traceability processes.

The thesis proposes factors and guidelines for traceability maintenance, for traceability tool developers and companies acquiring traceability tools. The feasibility of these guidelines are shown by implementing a traceability tool that is configurable and supports diverse artifacts and tools. To support the transfer of automated techniques of creating trace links to industry, we provide insights and lessons learned on improving the trace link vetting process. Lastly, the thesis proposes a traceability introduction methodology, which consists of concrete steps for companies to design, deploy and evaluate traceability strategies.