ABSTRACT

This dissertation tackles one of the most urgent problems in today's information technology, namely the renovation and migration of legacy information systems to modern platforms and net-centric architectures. In this context, several methods, tools, and processes have been proposed to support reengineering and modernizations of legacy database applications. This can be a complex task because many legacy databases have grown over several generations of programmers and lack a sufficient documentation. Computer-aided reengineering methods and processes have a great potential to reduce the complexity and risks involved in database design recovery and migration projects. Still, current reengineering tools are hardly adopted for practical problems in industry because they often make idealistic assumptions about the structure of legacy systems and the characteristics of reengineering processes. The goal of this thesis is to provide concepts and techniques to overcome these severe limitations. In particular, our focus is on developing mechanisms to manage uncertainty and inconsistency in computer-aided databases reengineering processes. In practice, uncertain knowledge plays an important role in activities aiming to recover conceptual design documents for large idiosyncratic implementation structures. This fact is neglected in current database reengineering methods and tools.

In this dissertation, we identify and extend a theory that provides a suitable basis to deal with uncertain reengineering knowledge and allows to implement practical tools and environments to support reengineering processes. The requirement for consistency management considers the fact that it is unrealistic to presume that database reengineering processes can be executed in a number of sequential phases or steps without iterations. In practice, larger reengineering projects comprise many process iterations due to various reasons like incomplete knowledge about legacy implementation structures or necessary "on-the-fly" modifications of the legacy system. Detecting and removing inconsistencies caused by such iterations significantly increase costs and durations of current reengineering projects. In this thesis, we employ graph transformation theory to develop mechanisms which allow to detect and eliminate inconsistencies between legacy schema implementations and their abstract representation, automatically. Our results have been implemented in the database reengineering environment *Varlet* and evaluated with an industrial project. They are suitable to complement many existing approaches in the domain of information system reengineering and migration. As an example, we describe the integration of *Varlet* with an existing middleware product for data integration.