An Overview of the
Temporal Query Language TQuel

Richard T. Snodgrass

TR 92-22

August 20, 1992

Abstract

We provide an overview of the previous decade of research on the temporal query language TQuel. TQuel is a minimal extension to Quel, the query language for Ingres. TQuel supports valid time and transaction time. Unlike many other temporal query languages, it supports aggregates, valid-time indeterminacy (where it is not known exactly when an event occurred), database modification, and schema evolution. We discuss all of these aspects, first informally through examples, then formally by presenting their tuple calculus semantics.

We start with the language itself, building incrementally from the core constructs to the more advanced features, in almost three dozen example TQuel statements. We follow the same approach in presenting the formal semantics. We present tuple calculus equivalents both for generic TQuel statements and for several examples.

TQuel is based on the predicate calculus. To execute a query, a procedural equivalent is required. We define a temporal algebra, again incrementally, and give several important properties, such as closure, completeness, and reducibility to the snapshot algebra. We also show how each TQuel statement can be mapped into the algebra. Finally, we discuss two important topics in implementing the temporal algebra: query optimization, specifically the applicability of existing optimization strategies, and the physical structure of pages storing temporal tuples. The paper ends with a BNF syntax of TQuel that incorporates all of these features.

This will appear as a chapter in the forthcoming book, *Temporal Databases: Theory, Design, and Implementation*, Benjamin/Cummings Pub. Co. 1993.

Department of Computer Science
University of Arizona
Tucson, AZ 85721
`rts@cs.arizona.edu`

An Overview of the Temporal Query Language TQuel