An Overview of Transaction Logic

Anthony J. Bonner*
Department of Computer Science
University of Toronto
Toronto, Ontario M5S 1A4, Canada
bonner@db.toronto.edu

Michael Kifer†
Department of Computer Science
SUNY at Stony Brook
Stony Brook, NY 11794-4400, U.S.A.
kifer@cs.sunysb.edu

Appears in *Theoretical Computer Science (TCS)*, 133:205–265, 1994.

Abstract

This paper presents an overview of *Transaction Logic*—a new formalism recently introduced in [11, 12] and designed to deal with the phenomenon of state changes in logic programming, databases, and AI. Transaction Logic has a natural model theory and a sound and complete proof theory. Unlike many other logics, however, it is suitable for *programming* procedures that accomplish state transitions in a logically sound manner. Transaction logic amalgamates such features as hypothetical *and* committed updates, dynamic constraints on transaction execution, nondeterminism, and bulk updates. Transaction Logic also appears to be suitable as a logical model of hitherto non-logical phenomena, including so-called *procedural knowledge* in AI, and the *behavior* of object-oriented databases, especially methods with side effects.

This paper is available at the following URL:
`ftp://db.toronto.edu/pub/bonner/papers/transaction.logic/tcs94.ps.gz`

*Work supported in part by an Operating Grant from the Natural Sciences and Engineering Research Council of Canada and by a Connaught Grant from the University of Toronto.

†Supported in part by NSF grant CCR-9102159.