Transaction Logic: An (Early) Exposé*

Anthony J. Bonner† and Michael Kifer‡

University of Toronto
Department of Computer Science
Toronto, Ontario M5S 1A4
Canada

Abstract

This paper is an informal account of *Transaction Logic*—a new logic recently introduced in [4]. Transaction Logic was devised to deal with the phenomena 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* transactions, transactions that accomplish state transitions in a logically correct 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.

---

*Keynote address in *Proceedings of the Workshop on Formal Methods in Databases and Software Engineering*, Springer-Verlag, Workshops in Computing Series, May 1993. Held, 15–16 May 1992, Montreal, Canada.

†email: bonner@db.toronto.edu. This work was supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada and a Connaught Grant from the University of Toronto.

‡email: kifer@cs.sunysb.edu. This work was supported in part by NSF grant CCR-9102159 and a grant from New York Science and Technology Foundation. Work done while on sabbatical leave from Stony Brook University. This author wishes to thank Computer Systems Research Institute of University of Toronto for their support. Present address: Department of Computer Science, SUNY at Stony Brook, Stony Brook, NY 11794, U.S.A.