Provenance for Database Transformations

Val Tannen

University of Pennsylvania
val@cis.upenn.edu

Database transformations (queries, views, mappings) take apart, filter, and recombine source data in order to populate warehouses, materialize views, and provide inputs to analysis tools. As they do so, applications often need to track the relationship between parts and pieces of the sources and parts and pieces of the transformations' output. This relationship is what we call database provenance.

This tutorial presents an approach to database provenance that relies on two observations. First, provenance is a kind of annotation, and we can develop a general approach to annotation propagation that also covers other applications, for example to uncertainty and access control. In fact, provenance turns out to be the most general kind of such annotation, in a precise and practically useful sense. Second, the propagation of annotation through a broad class of transformations relies on just two operations: one when annotations are jointly used and one when they are used alternatively. This leads to annotations forming a specific algebraic structure, a commutative semiring.

The semiring approach works for annotating tuples, field values and attributes in standard relations, in nested relations (complex values), and for annotating nodes in (unordered) XML. It works for transformations expressed in the positive fragment of relational algebra, nested relational calculus, unordered XQuery, as well as for Datalog, GLAV schema mappings, and tgd constraints. Specific semirings correspond to earlier approaches to provenance, while others correspond to forms of uncertainty, trust, cost, and access control.

This is joint work with J.N. Foster, T.J. Green, Z. Ives, and G. Karvounarakis, done in part within the frameworks of the Orchestra and pPOD projects.