Reminiscences on Influential Papers

Kenneth A. Ross, editor

I'm happy to be able to share with you the following three reminiscences. I continue to invite unsolicited contributions. See `http://www.acm.org/sigmod/record/author.html` for submission guidelines.

Yannis Ioannidis, University of Athens, `yannis@di.uoa.gr`.

[S. Christodoulakis. Implications of Certain Assumptions in Database Performance Evaluation. ACM TODS, Vol. 9, No. 2, June 1984, pp. 163-186.]

My personal interest in database estimation and approximation problems did not start until I read this paper a few years after it was written. I believe there are two reasons why it had such a significant influence on my work: its contents and its elegance. It took a comprehensive look at all the basic assumptions that database optimizers used at the time (and some still do) and studied their impact, showing that they generate estimates that are extreme while reality is often better behaved. Several other papers had approached similar subjects, but they either addressed special cases or were rather informal and empirical. I felt that this paper cleared the field by giving consise answers to several critical problems, and set the stage for much of the subsequent work in the estimation/approximation area. In addition to its technical merits from a database perspective, however, I was also attracted to its really elegant mathematics. This paper introduced me to Schur functions and the theory of majorization, and pointed me to the book of Marshal and Olkin "Inequalities: Theory of Majorization and its Applications," which was a constant companion in my later work on histograms both with Christodoulakis and others. It also gave ample evidence of the effectiveness of applying the underlying mathematical theories to "ugly" database issues. For me, this paper represents one of the most significant fertilizations of databases with "classical" mathematics, having impact on practical problems.

Anant Jhingran, IBM Almaden Research Center, `anant@us.ibm.com`.

[David J. DeWitt, Robert H. Gerber, Goetz Graefe, Michael L. Heytens, Krishna B. Kumar, M. Muralikrishna. GAMMA: A High Performance Dataflow Database Machine. VLDB 1986: 228–237.]

It is clear that many of my Almaden/IBM colleagues (with papers on System R, R*, ARIES, QGM, Starburst, View Maintenance, GARLIC and Data Mining) and my advisor, Mike Stonebraker (with papers on INGRES and POSTGRES) have had tremendous influence in the database field, and have basically taught me most of what I know in this area. However the paper which was my inspiration for five years (my early part of my IBM career) was by DeWitt et al. – "GAMMA: A High Performance Dataflow Database Machine," in Proceedings of VLDB 1986. The simplicity of their design (commodity parts connected through commodity network) is a model for parallelism and scalability that we are still living with and benefitting from. Our own database work (leadng to DB2 Parallel Edition and subsequently DB2 UDB EEE) was totally inspired from the GAMMA work – in particular, our partitioning strategies, our table queue concepts and our join techniques, paralleled and extended early GAMMA work.

I do not necessarily agree with all the results of the paper – in particular, the assumption that all equi-joins can be conducted efficiently through repartitioned hash joins was not borne out in our experimentations (hence we went to great lengths in co-locating joins where possible). In addition, other SQL constructs like subqueries pose fundamental headaches for parallel processing that GAMMA papers did not address. However, an excellent paper is one that challenges the readers to follow the same path and try to do one better – and this paper stands out in that regard.

Christos Papadimitriou, University of California, Berkeley, christos@cs.berkeley.edu.

[K. P. Eswaran, J. N. Gray, R. A. Lorie, and I. L. Traiger. The Notions of Consistency and Predicate Locks in a Database System. Communications of the ACM, 19(11): 624–633, 1976.]

When I was a first-year assistant professor at Harvard working in Theory, Phil Bernstein handed me a copy of the CACM article by Eswaran, Gray, Lorie, and Traiger. The "predicate locks" in the title and the emphasis on the "phantom" problem confused me only for a short while. The database concurrency control problem first identified in that paper captivated me as a rich, clean, and deep problem that was very specific to Databases, quite distinct from its relatives in Operating Systems and Concurrent Programming. Although many people would argue today that the concurrency control problem was analyzed and theorized about beyond its true importance and position on the critical path to database systems, for me it was crucial because it introduced me to the broader field of databases and to what later became the PODS community.