Reminiscences on Influential Papers

Kenneth A. Ross, editor

I would like to start by thanking Rick Snodgrass for introducing this column into SIGMOD Record. Since its inception, it has been a must-read for me each issue. I am delighted to be able to take over the job of editing such a well-liked column.

This column celebrates the process of scientific inquiry by examining, in an anecdotal fashion, how ideas spread and evolve. Following Rick's example, I've asked a few well-known and respected people in the database community to identify a single paper that had a major influence on their research, and to describe what they liked about that paper and the impact it had on them. Already I'm finding that the contributors really enjoy writing the paragraphs, because it gives them a moment to pause and think about their most heartfelt observations about their chosen field.

While I will continue to invite database researchers to contribute to this column, I also invite unsolicited contributions. I will select the most interesting of these for publication. The only prerequisite is that the author has published a paper in a refereed database conference or journal. See `http://www.acm.org/sigmod/record/author.html` for contribution guidelines. Send contributions to `kar@cs.columbia.edu`.

Surajit Chaudhuri, Microsoft Research, `surajitc@microsoft.com`

[P.G. Selinger, M. Astrahan, D. Chamberlin, R. Lorie, T. Price. Access Path Selection in a Relational Database Management System. *SIGMOD* 1979, Boston, MA, pp. 23-34.]

Soon after joining HP Labs, I started working on the problem of optimizing queries with user-defined predicates in the context of the Papyrus Project. In order to better understand the issues, we decided to do a quick and dirty implementation of a System-R style optimizer. The experience of implementation, and discussions with Waqar Hasan, Ravi Krishnamurthy and Kyuseok Shim helped me understand in depth the significance of key aspects of the System-R optimizer. During the period 1992-1995, along with my colleagues at HP Labs, I worked on the problems of optimizing queries with group-by, user-defined predicates, and materialized views. In addressing each of these problems, we found ourselves leveraging the generality of the ideas such as interesting orders, use of dynamic programming, and choosing among comparable intermediate plans.

The System-R optimizer was noteworthy for defining the architecture of relational query optimization systems and many essential algorithmic elements. At a more personal level, it taught me to worry not only about transformations, but also about the impact of the transformation on the rest of the optimizer; and how best to leverage transformation rules within an existing query optimizer without increasing optimization time. It would be a worthwhile effort to develop and elucidate the principles of optimization of multi-block SQL queries with the same clarity as System-R did for SPJ queries. Admittedly, it is a tall task.

Gösta Grahne, Concordia University, grahne@cs.concordia.ca

[Tomasz Imielinski, Witold Lipski Jr. Incomplete Information in Relational Databases. *JACM* 31(4):761–791, 1984]

In the early eighties I was an undergraduate student at the University of Helsinki. This was in the middle of the Cold War. The placement of Finland in the East/West partition was a bit fuzzy, so Helsinki was one of the few Western places academics from the East block could easily visit. In Helsinki we had a steady stream of visitors from the Soviet satellites on various "friendship" exchange programs. Nobody paid much attention to these visitors. However, I heard one day that one them was working on "null values," something which I was interested in. So I went to talk to Tomasz Imielinski, a "friendship" visiting graduate student from Warsaw. Tomasz explained the *tables*-construct to me, and I was sold. Meanwhile, as Jeff Ullman and Alberto Mendelzon describe in their reminiscences, the *chase* and an arsenal of other tools were being developed at Princeton.

Later I wrote my PhD-thesis on tables, and worked on incomplete information with several database people. Today, incomplete information is experiencing a renaissance, due to its central role in information integration applications. Serge Abiteboul's recent comment (PODS'99, invited talk) that "[tables] are a typical representative of great tools that remain unfortunately mostly unused" is somehow emblematic of the story.

H. V. Jagadish, University of Michigan, jag@eecs.umich.edu

[Jim Gray and Franco Putzolu. The 5 Minute Rule for Trading Memory for Disc Accesses and The 10 Byte Rule for Trading Memory for CPU Time. *SIGMOD* 1987, San Francisco, CA, pp. 395-398.]

This paper makes a very simple point. Based on the relative prices of disk arms and semiconductor memory in 1986, it made sense to apportion the budget allocated to purchase storage in a computer system such that exactly the items likely to be accessed more frequently than once every 5 minutes could be memory resident. Similarly, it was worth spending 10 bytes of memory to store partial results if it saved one CPU instruction. Of course there are many caveats to any rule as simple as these, some of which are addressed in the paper (such as the variation in data item size), and others of which are not even mentioned (such as the value to be placed on faster response enabled by memory-resident data, and such as temporal variations in access frequency).

Clearly, the specific cost numbers from 1986 are not applicable any more. Nonetheless this paper is amazing in several respects. It runs only three and a half pages. Once the basic idea is established, there is little attempt to "fill out" the paper with extraneous technical material. The central notion of economic equivalence in choosing a system configuration is something every computer science undergraduate should (and could) be taught. It is a wonder this paper made it through the usually conservative SIGMOD program committee.

SIGMOD 1987 was the first database conference I attended, at a time when I was inclined to build a career in databases, but not yet quite sure about my choice. This paper was central to cementing my decision to work in the database area. At a juncture when I was disappointed by the lack of applicability of research work (in other areas) I had been engaged in up to that point, this was a blast of fresh air. And the central lesson of the paper will stay with me for life – quick economic analyses are an essential step in system design.

Jan Van den Bussche, University of Limburg, vdbuss@luc.ac.be

[S. Abiteboul and P.C. Kanellakis. Object identity as a query language primitive. First presented at *SIGMOD* 1989. Full version appeared in *J.ACM*, 45(5):798–842, September 1998.]

I am a theoretician, and most of my publications have been on trying to understand exactly what you can do, and what you cannot do, with various query languages for various data models. When Ken Ross asked me about a particular paper that had a big influence on me, the paper on the Identity Query Language (IQL), by Serge Abiteboul and Paris Kanellakis, immediately started blinking inside my head. I still remember vividly, one day in the autumn of 1989, how Jan Paredaens, my advisor at the time, threw the INRIA technical report on my desk and suggested that this could be interesting. Was he right.

The IQL paper is perhaps best known for its definition of a most elegant data model for object-oriented databases with complex objects, object identity, classes, types, and subtypes. However, the paper also deals with trying to understand the expressive power of query languages that allow you to introduce new object identifiers in the result of a query. Exactly that topic turned out to be also the topic of my PhD dissertation a few years later, so the influence of this paper on me personally will be immediately clear. However, what is equally important is that the paper rests on a rich foundation laid by earlier, seminal work on the theory of database queries: equally influential papers come up here, such as those by Chandra and Harel, by Vardi, and by Abiteboul and Vianu. In the ten years since 1989, the theory of database queries has continued to advance, also with the help of the finite model theory community. Still, the IQL paper remains a truly deep paper and my personal favorite.

Moshe Vardi, Rice University, vardi@cs.rice.edu

[D, Maier, A.O. Mendelzon, Y. Sagiv. Testing Implications of Data Dependencies. *ACM TODS* 4(4):455-469, 1979]

I can trace the start of my research career to this paper. In 1980, I was a master's student at the Weizmann Institute of Science, Rehovot, Israel. The paper was given to me by Catriel Beeri to help me prepare my presentation in a graduate seminar on theorertical computer science. I remember being captivated by that paper. Unlike some of the earlier papers in database theory that I have read by then, this paper did not seem ad-hoc. Rather it defined the implication problem for dependencies as a fundamental problem in database theory and went on to develop a methodical approach, called "the chase", for testing such implications.

The question posed at the end of the paper regarding implication of join dependencies was the motivating question for my master's thesis, and my doctoral thesis can be viewed as an effort to generalize the chase method developed in the paper and identify its limits. The DBLP bibliography lists 74 papers that cite the paper that we used to describe as the "MMS paper". This is clear evidence to the strong role that this paper played in database theory in the last 20 years.