

Towards a Science of Temporal Databases

Richard T. Snodgrass
Department of Computer Science
University of Arizona
Tucson, AZ
rts@cs.arizona.edu

Abstract

Computer science has long been considered to emphasize three distinct perspectives: mathematics, science, and engineering. While the database field has some very strong mathematical and engineering work, the scientific perspective has been much less prominent. This keynote will elaborate the scientific perspective, apply it to research questions in temporal databases, and emphasize that this new way of thinking can yield new insights and deeper understanding.

1. Introduction

Three quite distinct perspectives: mathematics, science, and engineering, have long been associated with the discipline of computer science (CS) [Denning 2005]. Mathematics appears in CS through mathematical formalism, mathematical theories, and algorithms, the latter of which are ultimately mathematical objects that can be then expressed as programs. Engineering, which is concerned with making things better, faster, smaller, and cheaper, is prevalent in CS. Science may be defined as developing general, predictive theories that describe and explain observed phenomena, and evaluating these theories [Aicken 1984, Chalmers 1999]. Theories include a statement of “why” and are amenable to predictions on heretofore unexamined phenomena, that can be subsequently tested on those phenomena [Davies 1973].

Nobel Prize winner Herbert Simon identified CS as a “science of the artificial,” strongly asserting, “the computer is not just an instrument but a phenomenon as well, requiring description and explanation” [Newell, Perlis, and Simon 1967, page 1374]; he furthered that line of reasoning in his influential book [Simon 1996]. However, the promise of a science of computational phenomena has not yet been realized by this fortieth anniversary of Simon’s compelling argument.

2 Science in Database Research

While the database field has some very strong mathematical and engineering work, there is to date very little true science that has been done. One common scientific methodology is hypothesis testing. Of the more than 10,000 papers (150K pages) in the SIGMOD Anthology, capturing most of the database papers published in the last twenty-five years (1975–2000), there were all of 37 papers that even *mentioned* hypothesis testing. Of these, less than a dozen actually used this methodology. A rather dated analysis of 57 journal articles on multi-key search algorithms [Eastman 1983] found that “statistical tests for hypothesis testing were used in only two papers.”

As another cut, experimentation is important in science [Denning 1980]. A quantitative study of a random sample of refereed CS publications (SIGMOD papers comprised 18% of these papers) found that “of the papers in the random sample that would require experimental validation, 40% have none at all. ... In comparison, the fraction of papers lacking quantitative evaluation in *OE* [*Optical Engineering*] and *NC* [*Neural Computation*] is only 15% and 12%. The low ratio of validated results appear to be a serious weakness in computer science research.” [Tichy et al. 1995, page 9].

The philosophy of science, including the scientific method, provides an opportunity to apply new approaches, new tools, and new forms of reasoning to the fundamental problems confronting temporal databases and CS in general.

3 Benefits of the Scientific Perspective

Science seeks to explain phenomena. These explanations take the form of falsifiable *scientific theories* [Popper 1969] and *scientific laws* [Achinstein 1971].

General theories are preferred over idiosyncratic explanations, for three primary reasons.

- *Parsimony*: Successful theories explain a wide variety of phenomena with a short, coherent explanation (Occam's razor).
- *Prediction*: Successful theories can predict future events, often to a high degree of accuracy.
- *Control*: Theories can be used to ensure positive effects or avoid negative effects.

Mathematics is often useful in providing formalisms to succinctly express theories (parsimony). Science can be useful to engineering by explicating the underlying causal mechanisms (prediction and control). Similarly, engineering reveals behavior (phenomena) that science can use to construct new theories, and science provides needs for new formalisms and theorems. Science provides a specific methodology, the scientific method, which has been extremely productive and beneficial in other sciences. The development of scientific theories in CS generally and in database systems specifically can produce new insights and better outcomes than restricting ourselves to the mathematical and engineering perspectives. Denning's recent assessment is hopeful: "The science paradigm has not been part of the mainstream perception of computer science. But soon it will be" [Denning 2005, page 31].

4 Science in Temporal Databases

This keynote address will examine the relative roles that the mathematical, scientific, and engineering perspectives might play specifically in the study of temporal databases, as an exemplar for CS in general. We will examine how the notions of falsifiability, theory, laws, paradigms, and research programs apply in the philosophy of science. We will consider several scientific theories that have been advanced in CS. We will then outline areas within temporal databases where the scientific perspective might yield new insights, theories, and laws. This new (to CS and to database research) way of thinking has the potential to fundamentally alter how we organize, view, retrieve, store, and generally manage time-varying data, and indeed, how we fundamentally understand those software systems that contend with such data and the people that use those systems.

Unlike other established sciences, where many if not most of the fundamental theories have already been discovered [Horgan 1996, Horgan 2004] (though much elaboration remains), the theories of temporal databases and of computer science in general are still out there, just waiting to be uncovered.

5 Acknowledgments

This research was supported in part by NSF grants IIS-0415101, IIS-0639106, and EIA-0080123 and with partial support from a grant from Microsoft Corporation.

References

- [Aicken 1984] F. Aicken, **The Nature of Science**, Heinemann Educational Books, London, 1984.
- [Achinstein 1971] P. Achinstein, **Law and Explanation**, Clarendon Press, Oxford, 1971.
- [Chalmers 1999] A. F. Chalmers, **What is this thing called Science?** Third Edition, Hackett Publishing Company, 1999.
- [Davies 1973] J. T. Davies, **The Scientific Approach**, Academic Press, New York, 1973.
- [Denning 1980] P. J. Denning, "What is Experimental Computer Science?" *CACM* 23(10):543–544, October, 1980.
- [Denning 2005] P. J. Denning, "Is Computer Science Science?," *CACM* 48(4):27–31, April 2005.
- [Eastman 1983] C. M. Eastman, "Current Practice in the Evaluation of Multikey Search Algorithms," in *Proceedings of the SIGIR Conference*, pp. 197–204, 1983.
- [Horgan 1996] J. Horgan, **The End of Science**, Helix Books, Addison-Wesley, Reading, MA, 1996.
- [Horgan 2004] J. Horgan, "The End of Science Revisited," *IEEE Computer* 37(1):37–43, January 2004.
- [Newell, Perlis, and Simon 1967] A. Newell, A. J. Perlis, and Herbert A. Simon, "Computer Science," letter in *Science* 157(3795):1373–1374, September, 1967.
- [Popper 1969] K. R. Popper, **Conjectures and Refutations**, Routledge and Kegan Paul, London, 1969.
- [Simon 1996] H. A. Simon, **Sciences of the Artificial**, Third Edition, MIT Press, 1996.
- [Tichy et al. 1995] W. F. Tichy, P. Lukowicz, L. Prechelt, and E. A. Heinz, "Experimental Evaluation of Computer Science: A Qualitative Study," *J. Systems Software* 28:9–18, 1995.