Remaining CALM in Declarative Networking

Frank Neven
Hasselt University and transnational University of Limburg
frank.neven@uhasselt.be

ABSTRACT

Declarative networking is an approach where distributed computations are modeled and programmed using declarative formalisms based on extensions of Datalog. On a logical level, programs (queries) are specified over a global schema and are computed by multiple distributed computing nodes over which the input database is partitioned. These nodes can perform local computations and communicate asynchronously with each other via messages. The model operates under the assumption that messages can never be lost but can be arbitrarily delayed. An inherent source of inefficiency in such systems are the global barriers raised by the need for synchronization in computing the result of queries.

This source of inefficiency inspired Hellerstein to formulate the *CALM-principle* which suggests a link between logical monotonicity on the one hand and distributed consistency without the need for coordination on the other hand (CALM stands for Consistency And Logical Monotonicity). A crucial property of Datalog programs (and monotone programs in general) is that derived facts must never be retracted when new data arrives. The latter implies a simple coordination-free execution strategy: every node sends all relevant data to every other node in the network and outputs new facts from the moment they can be derived. No coordination is needed and the output of all computing nodes is consistent. This observation led to the CALM-conjecture which, in its revised form[1], states

> *"A query has a coordination-free execution strategy iff the query is monotone."*

In this talk, we discuss the CALM-conjecture (what does it even mean to be coordination-free?) and its status: when computing nodes are increasingly more knowledgeable on how tuples are distributed, increasingly more queries can be computed in a coordination-free manner. In fact, these classes can be characterized in terms of increasingly weaker forms of monotonicity thereby adding a new dimension to the CALM-conjecture. Furthermore, we discuss novel (and more traditional) fragments of Datalog (with negation) capturing the above classes. At the end, we point out future directions and challenges for the community.

[1]The original conjecture replaced monotone by Datalog.

Categories and Subject Descriptors

H.2.4 [**Information Systems**]: Database Management—*Systems, Distributed Databases*

General Terms

Languages, Theory

Keywords

Datalog, monotonicity, relational transducer

10.5441/002/icdt.2014.02