The different effects of the database update during the long execution of continuous queries

Antal Búza

The continuous data streams are native data occurrences. Perhaps ones are more native than the relations in lot of cases. Thus the using of the data streams may have number of advantages. Therefore the expansion of CQL, since being suitable for really situations might be useful. This is the reason for the paper suggests a few expansions of CQL.

The CQL, Continuous Query Language is an expressive SQL-based declarative language for registering continuous queries against streams and updatable relations. CQL is suitable for data stream queries. There are situations when the queries operate on relational databases and on the data streams simultaneously. The execution of CQL query takes a long time (it may be several hours, days or even more). It is not unambiguous which semantic is suitable for the user when the database is updated during the execution of CQL query. For example, when we wish keep an eye on changing the value of an account while the official rates are updated, then CQL system must calculate with the retroactive effect of this update. Another semantic is reasoned when the prices are changed while we use CQL query for observing the trade of a supermarket. In this case the effect of the update is valid from the moment of update. In this paper we give a short description of CQL, characterisation of update-problems, and we offer possible suggestions for the semantically extension of CQL.

Additional interesting question is the explanation of the consistent state. In classical database-theory it is an usually requirement that the "normal" state of a database is the consistent state. The consistency is not a permanent state, during the updates it may be damaged for a short time. Investigations were performed to find out what the effect of the inconsistent state on the currently operating CQL queries is.