Transactions on replicated data in mobile Ad Hoc nets – and some further issues of large scale networks

Heinz F. Schweppe
Institut f. Informatik, FU Berlin
hs@inf.fu-berlin.de

A transactional Replication System

In the Coco/Da project[1] data centric middleware functions for Mobile Ad Hoc Nets have been investigated and developed. Transactional support[2] and data replication are important functions of traditional data management systems. Using a volatile communication infrastructure, replication plays an even more important role than in fixed wired systems in order to increase availability of data. Keeping replica consistent, however, is more involved with unreliable communication.
Data are fully replicated in our approach on each node. A lazy update propagation strategy guarantees eventual consistency at all nodes. Updates may be transactions: each transaction is executed locally and lazily propagated to other nodes. Transactions that have been executed locally and shipped to other nodes are in the precommitted state. If no conflicts are detected they may become committed. Conflicts are detected without any central control. The utilization of appropriately adjusted physical timestamps and the shipping of the total sequence of operations of transactions allows for conflict detection and resolution in a lightweight manner. Arbitrary conflict resolution strategies may be plugged in, e.g. based on commit time, transaction begin or semantic criteria. The protocol not only allows for eventual consistency but also for one-copy serializability.

An important aspect of this type of TA protocols is effectiveness. If there are too many conflicts, a protocol which keeps replica consistent, turns out to be useless. However in applications like group collaboration with low transaction rates, the danger of conflict is rather low. We implemented a mobile kind of Wiki. Users cooperatively write articles. In case of low transaction rates per user – e.g. a few per day per node – the danger of a conflict is very low. For details see:
Scholz, Bregulla, Schweppe.: *Symore – A replication system for Mobile Networks*, FU Berlin, 2006 (submitted for publication).

Some remarks on Evolving Networks – or: do we solve the right problems?

There is no doubt that large scale networks of computing devices of any size – from sensor nodes up to high end computers – will play a dominant role in the next decade. Most of the nodes, in particular the small sized, will have a high degree of autonomy, e.g. they may join or leave the community at any time. This is in particular true in those parts of the net which are not fixed wired. In this case communication breakdown is not an exception but has to be viewed as part of normal processing.

On the other hand *computational guarantees* like determinism or atomic commit are most important for traditional computational processes. Therefore it is no surprise that much effort has been put into making computation in unstable networks reliable. Many achievements have been made within the past few years. However it becomes more and more clear, that many applications in large scale networks are of a best-effort type. Transactional guarantees are not only hard to achieve in mobile networks, but they are not even required in many application scenarios.[3]

[1] Funded by DFG
[2] Transactions without full replication are discussed in the contribution of Joos Böse
[3] There are scenarios like E-Commerce, where transactions are undoubtedly important.

It is therefore a natural question if the perspective on computation and information management in large scale networks, which has been taken by many researchers in the past, is the right one. At least it is not the only perspective. *Uncertainty* is a phenomenon which is ubiquitous in large scale networks with autonomous nodes. The simplest example is web retrieval. Search engines typically deliver a good approximation of the most relevant documents. However there is no guarantee, that the most relevant ones have been found. Data from different sources may be more or less reliable. This is why data lineage is an important issue. Data in the net may be inconsistent. If several sources support a particular fact ("*A's date of birth is d*") but some supply a contradicting value, an approximate probability may be calculated. Many questions arise: Are there any guarantees which are less rigid than transactional consistency? If there are many unstable nodes providing a particular service, can we learn statistical properties about the availability of the service? How can the freshness of data be estimated?

Standard applications which need traditional guarantees will be performed in stable environments also in the future. However, it is important to deal with uncertainty more explicitly in large, evolving network. This opens a new perspective on this type of network with many open research questions.