# Critical Review of “Data Management in the Cloud”

Zhiquan Sui

I just go through the content of the paper and write some comment in it.

This paper mainly states that Cloud computing is not proper for all kinds of data management. It is good for analytical database management which is not that write-intensive. The author mentions that “luckily” there are more and more analytical operations on database, so it seems that in some day most data management will be analytical and embedded in cloud. But in my opinion things are not that simple. The trend of cloud computing is more and more concentrated. It means the largest clouds should be kept by several particular companies such as Google, Amazon, Yahoo, and so on. The analytical operations are more and more because these companies are in good situation so that people just use their data which has been uploaded into cloud. However, the economics situation changes every time. It is hard to predict whether there will be a day that the cloud owners will have bankrupt so that people have to re-upload their data into some new cloud. I think at that time, the transactional data management will be majority of data management again.

After saying about differences between analytical data management and transaction data management, the author says something about cloud DBMS wish list versus current software in cloud computing. It seems that neither MapReduce-like software nor shared-nothing parallel databases satisfied everything in wish list. However, “luckily” again, except operation on encrypted data, either MapReduce-like software or shared-nothing parallel databases does satisfy the remaining items in wish list. So the author said that we can bring a hybrid solution which satisfies all wishes. But what’s important is that the hybrid is not a simple plus operation. For example, if we combine these two together, whether the better performance will dominate, or the performance will be delayed by the worse one. Also, if one does not adapt to heterogeneous system, whether the other one can help.

In the end of this paper, the author addresses a question which is how to balance the tradeoffs between fault tolerance and performance. I think it is a proper work for machine learning to find such a balance, although I have no sense for machine learning. But I think even if we set some options to users to choose, the users always have no sense about which is really a good option for their environment. Also the options cannot be that accurate, they are just some rough level. Another advantage for automatically doing that is the environment might be changing every time by the aging for the machines, not stable network situation and so on. The automatic balance can modify the parameters time by time which make sure that the parameter is proper to the current environment.