Supporting activity for Week four

Disadvantage of Normalizing to a high degree

Because the process of normalizing a database takes time, normalizing to a high degree is time consuming. The higher the normalization the more tables the database is going to have, therefore more files implies more memory to manage them. The amount of accesses to disk will increase directly proportional to the number of files in the scheme. The performance of the database system will suffer in response time. Also, it will require more time of database maintenance. Another disadvantage of normalizing to a higher degree is that it will need a bigger amount of joins for retrieving the information desired.

After the 3NF or 4NF it is time to answer the question, is it worth to keep normalizing? The cost-benefit dilemma of normalizing a database is another way of questioning the outcomes of this process at the time of talking to companies’ decision makers. A database well normalized it is not free. The storage space saving has a cost, and that cost is the loss in performance. There are some facts to considerate when one has to make the decision when to stop normalizing: are there more inserts than retrieves or it is the other way around? Is the application for banking in real time or time is not an important factor? Is the database environment centralized or distributed? Does the application commit with multiple phases or single phases? These and other relevant questions play an important role for making the decision of when to stop normalizing.

References

<http://technology.blurtit.com/595612/advantages-and-disadvantages-of-database-normalization>

<http://www.techrepublic.com/article/to-be-or-not-to-be-normal-that-is-the-database-question/#>.

<http://semanticdb.blogspot.com/2007/08/normalization-where-to-stop.html>