Andrew Zurn

Reading 8

Summarized Article: <http://www.drdobbs.com/article/print?articleId=240162452&siteSectionName=database>

Other Article: <http://cacm.acm.org/magazines/2013/10/168176-the-alan-turing-year-leaves-a-rich-legacy/fulltext>

Database systems and the widely used SQL language are nothing new, spanning decades back into some of the earlier years of computing. However, around 2010 a shift away from the traditional relational model of database systems has occurred, as new databases like couchDB, mongoDB, Postgres, among dozens others, have emerged, with a new querying system following them known as NoSQL. This new model strayed from tradition, and schema-less, widely distributed document-based databases were ushered in with the coming of the age of big data. Even with this new shift by many hip companies like Amazon, Google, Facebook, and others, SQL and relational databases still holds it's strong majority in the database world.

Seth Proctor, CTO of NuoDB examines the current trends of the database world, and tries to answer the question as to why SQL still has this hold on the database community. Most technologists know the relative history of SQL, which is based on the relational model that was developed in the 70's and a decade later was implemented as a querying language. Since then, SQL grew in popularity, as Proctor points out, it solved “concrete problems.” SQL was widely used in the industry, until recently when a new model was developed, that being the document-based “NoSQL” model, which debuted in “2009”. This approach also solved a very concrete problem, that being how data could be effectively distributed across long distances as large application systems started to scale-out. NoSQL, effectively, “sacrificed consistency for scalability,” Proctor exclaims. However, Proctor goes onto states “the last few years have seen renewed excitement around SQL. NewSQL systems have emerged that support transnational SQL, built on original architectures that address scale-out requirements.” In the end, as he goes onto proclaim, what this new shift “means for the evolution of these systems is yet to be seen, but clearly, the appeal of Codd's [relational] model is as strong as ever 43 years later.”

I find these shifts between the traditional relational and document-based models quite interesting, as they both solve the problem of persistent storage, although they do it in very different ways. Although I'm apt to say I do enjoy the structure of the relational model when it comes to storing data and querying for that data, having recently done some research into a NoSQL database, couchDB, I do see the usefulness of these system, especially when in comes to storing terabytes of data across hundreds of different servers. It is quite intriguing hearing that there has been a new shift towards this NewSQL approach that has apparently solved the problem of scalability, and will definitely remain an area of technology to watch in the coming years as data becomes even larger.