Big Memory, Big Data, and the Semantic Web

Posted by Mitchell Shults on Jul 6, 2011 10:15:54 AM

Perhaps you've noticed that there are some fairly large systems being built using the Intel E7 Xeon processor. HP, Fujitsu, Supermicro, NEC, Bull, SGI, and many others have either announced new 8-socket (or larger) designs, or described their intentions to do so. You may be asking yourself, "Are there really enough customers for such systems to make the market interesting?"

After all, some would have you to believe that there is no large data-analysis problem that can't best be solved with a pile of inexpensive dual-socket servers, some Ethernet, and an army of Hadoop programmers.

And, there is certainly a class of problems for which this approach is just fine. These problems are perhaps best described as the 'small number of needles in a very large haystack' sort. Yet, there is another class of problems that is simply *not* well-suited to Hadoop-style processing.

As Franz Technology recently demonstrated at the Semtech conference, at least one of those problems appears to be the management of very large-scale, complex and rapidly-changing semantic database contents. I've only known about Franz for about 9 months now. They were first brought to my attention by colleagues at Amdocs, who were using Franz' technology to explore some really cool ideas (more on that shortly). Less than a month before the event, Franz approached Intel with a proposal.

"We'd like to show the world that the loading and querying of a trillion triples is possible on a large-scale Intel server platform - can Intel help?" â€“ asked Franz.

If you're anything like me, then you're probably wondering what the heck a 'triple' is in that sentence, and why you should care about the ability to load and query a trillion of them at once. I just had to satisfy my curiosity about this, so we found a way to get Franz some time on our big machine.

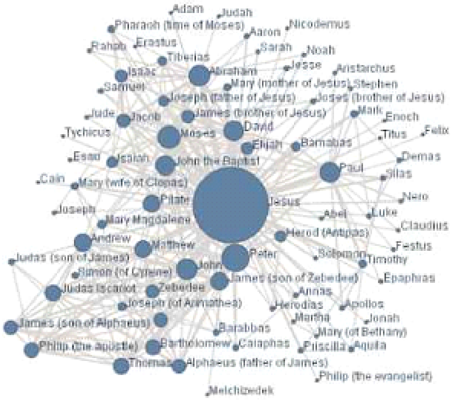
The term 'triple' is shorthand for the lowest level of data representation specified by RDF - the resource descriptor framework definition of the W3C. There's more to it, but the basic idea is to express any and all information in the form of subject->verb->object representation. An example of a single triple would be "Mitch", "Graduated From", and "Rice University".

My entire educational and work history could be expressed as a series of such triples. A Web-accessible database, such as a 'semantic' version of LinkedIn, could use a triplestore to support queries of work, educational history, geographic location, etc. that are far more accurate and sophisticated than mere full-text or keyword searches, which often produce nonsensical results and always require time-consuming human interpretation.

A relational database could be used to store my name, address, work history, educational history, etc. in a very storage and query-efficient form. However, someone has to design a relational database and someone has to maintain a relational database as the represented subject area undergoes change.

Relational databases are exactly the right way to deal with classic business structures and relationships - sales catalog items, warehouse inventory, shipping logistics, customers, orders, etc... And I'm not suggesting here that that's likely to change. But relational databases are very difficult to press into service for 'fuzzier' applications, such as characterizing the linkages in a social network or predicting why a caller on a customer service line might be calling before the service representative picks up the phone.

Consider the case of a social network - the sort of things LinkedIn, Facebook, and others are doing every day, at massive scale. Here's a representative social network, one that should be fairly recognizeable:



Just for this simple example, representing all of the relationships in this social network in a way that allows calculation of 'social distance' and gauging of relationship strength (two things that turn out to be important) requires many thousands of triples. Real-world examples rapidly become vastly more complex.

But Franz and their customers are demonstrating that triplestore approaches, thanks to the enormous power and capacity of large-scale enterprise class Intel Xeon E7 server platforms, in fact *are* practical (and cost-effective).

The history of computing is basically the story of finding creative ways to burn ever-less-expensive compute time in order to save ever-more-expensive programmer time. Semantic database techniques are the latest in a long list of innovations to advance generality and flexibility that are only practically possible thanks to Mooreâ€™s Law. At Intel, our job is to keep Mooreâ€™s Law on track so those innovations can keep happening..

Check back next week to find out if Franz Technology was able to pull it off!