24th June 2011

Hadoop and Lustre - Some Thoughts

Hadoop has become a major technology for new media companies that have to handle large textual data. Large hadoop clusters can have 1000's of nodes, making them to seem to be similar to a large HPC cluster. The difference is that in Hadoop clusters, the disk is local to each node, so that each node can process the data locally on that node. In HPC clusters, typically each node has limited disk space and most of the space is on a distributed, high performance file system like Lustre.

Storing HDFS data locally, means that hadoop is less sensitive to network speeds, and usually runs well on a 1gb network. But what happens if you run hadoop and put all of it's data into a distributed file system like Lustre? There have been several recent tests in doing that, essentially turning all or part of an HPC cluster into an hadoop cluster. Oracle Grid Engine now offers Hadoop Scheduling as part of its product. The HOD (Hadoop on Demand) project, adds Hadoop support to the Torque grid scheduling system. Initial results of this approach were a little discouraging, since Hadoop on local disks seemed to outperform hadoop on a Lustre file system, although it still performed pretty well. However, early tests were not on production HPC clusters, but were on small test clusters running a 1gb ethernet backbone for the Lustre file system. This basically limited the Lustre performance to be about the same as an 80MB/sec sata disk.

More recently tests [http://www.olcf.ornl.gov/wp-content/events/lug2011/4-12-2011/1100-1130_Nathan_Rutman_MapReduce_Lug_2011.pptx] have been done on an Infiniband network, which delivers much higher bandwidth. In this environment, Hadoop on Lustre starts to really perform, sometimes as much as 3 to 1 over local disks. I.E. the cluster only has to be 1/3 the size if it is running on an infiniband or other low latency, high bandwidth network fabric.

But there may be even more performance to be squeezed out of a Hadoop/Lustre configuration. In a Lustre file system, basically all data can be accessed by any node at the same high speed, typically at least 1 GB /sec. Just shoving Hadoop on top of Lustre doesn't take advantage of the fact that Hadoop is still going to be MOVING lots of data from one node to another, albeit at a very high speed.

This seems pretty stupid, since the data is just being moved from one location on the Lustre file system to another. A much better solution is to just have the hadoop namenode just update the file tables of the data nodes, telling them what data is now "local" to them. In this way, the data movement would not actually occur, just the pointers to the data. A study [http://arch.eece.maine.edu/superme/images/4/4b/Dunnmid.pdf] by a student at the University of Maine hint at this approach but didn't actually try it. A paper [http://wiki.lustre.org/images/1/1b/Hadoop_wp_v0.4.2.pdf] by interns at Sun actually tried it, but only on a 1gb network with a very small (2 OST) Lustre system

In fact, this approach should eliminate the Hadoop Lustre performance bottlenecks on a 1gb network, since data movement would be dramatically reduced. In fact, in the Sun paper, even a small Lustre file system performed well on a 1gb network.

Food for thought...

Posted 24th June 2011 by Norman White

Labels: fermi hpc, HADOOP, hadoop performance, lustre, Oracle Grid Engine

2 View comments

Anonymous May 29, 2012 at 4:31 PM

Hi Norman,

Very interesting topic that you have discussed above.But what is limiting companies in integrating Hadoop with Lustre if the performance benefit is so good with better networking bandwidth? Also can you provide a public link of the "study done by a student from University of Maine" ?Thanks

Reply

Sai Santosh October 26, 2015 at 4:54 AM

We never miss a single post on this blog about hadoop. After attending hadoop online training, this site worked as a supplement to our technical knowledge about the subject related to cloud and other related platforms like hadoop.

Reply

Enter your comment...

Comment as: The Bull Tamer (**Sign out**

Publish **Preview** □ Notify me