Possible follow on to PingERLOD project

The main issue on dealing with big data in an efficient way is that we need High Performance Computing (HPC). It is hard to perform specific complex ad-hoc analytical queries using current technologies if we do not use a cluster or something similar (e.g., VMs on a cloud provider, such as AWS).

An example of a specific ad-hoc query would be: How do all 16 metrics PingER measures behave from 2015-04-25 11:00:00 to 2015-04-25 15:00:00, from SLAC to Nepal compared with SLAC to all other cities in Asia, aggregating these other cities by country, in the same time interval?

Wouldn't it be interesting if we had a Web Application which we could write specific queries like this? The data available would be all data, in hourly grain, until the previous day. Then, this Web App would issue the query to the distributed database which returns the result set (which is obviously much smaller than the entire data) to the user. We would need to think about usability issues. We probably don't want to let many people perform multiple queries at the same time because the system may crash.

Indeed, it would be even better if the entire data, not only the result set, were in RDF linked open format. However, to the best of my knowledge, there are no freeware technologies that efficiently deal with big Linked Open Data entirely. There are lots of research efforts on it, though. There are also some companies that sell similar technologies, but they are usually very expensive and, to deal with larger datasets, they also require a distributed environment.

Hadoop or whatever other current technology that perform big data analysis work on a distributed environment. For this reason, if we really want to perform complex interesting ad-hoc analyses on PingER entire big data, ranging from 1998 until current days in hours, we need HPC, i.e., a cluster or similar. Otherwise, we will need to reduce the dataset we are going to deal with, as I had to do in my work while staying at SLAC, which may not be a problem.

As Thiago said, the main challenge is first to find such a distributed environment. This is actually our main problem at UFRJ at this moment. Then, after finding it, install the necessary technologies and set the environment, which is also not trivial. After having all this set and executing a first "Hello World" example on these technologies, everything else tends to be easier.

Something else: most of these technologies I have been referring to, including Hadoop, require a shared nothing file system. That means that even though the nodes in the cluster may work on a shared file system, they need some local storage which is not shared among the other nodes.

Les and Bebo, maybe you could ask Jacek Becla if he knows of some cluster we could use at SLAC. For our purposes, I believe 4 nodes with reasonable RAM memory each (16GB?) and lots of hard disk storage (at least 500GB~ in total?) would do the job.