Background:

We’re a data & visualisation company. We’ve built a distributed database system on top of postgresql that currently serve over 100 customers in Europe, Middle east & South America. Our largest installations process and store 4TB (10^12 data points) per day, total data managed being measured in hundreds of terabytes.

Assignment:

With standard postgresql tables and indexes, working with time series data can be quite heavy, especially when inserting data in a big table (100 million rows) with standard indexes on relevant columns. Our data is structured as "time", "key columns" and "data columns". Key columns are “meta data” used for grouping. Data columns are the actual measurements which in general do not have any index.

Currently we have the following properties for online data insertions:
• The time always increases. That is, we do not insert data for the past
• The combination of the time column and all key columns is always unique

We are interested in an index that can be made more efficient considering these properties. It should still be efficient to search on the key columns, both for aggregation and finding individual entries. The properties of the index that we are hoping to achieve with this:
• Constant time insert.
• Same performance as standard B-tree index in postgresql.

Performance metrics for testing the solution should include insert-performance, lookup-performance and aggregation-performance, both on time and on key columns. The implementation should be done as a standalone extension to postgresql.

Interested?

Interested? Contact
http://www.agama.tv/about-agama/careers/research-and-development/
aner.gusic@agama.tv