Automaatio^{XXI} 2015 Open source databases in industrial applications

Dmitriy Kuptsov

Ibisense, Sinimäentie 10 A, 02630 Espoo, Finland

Tel: +358 44 355 35 24, E-mail: dmitriy@ibisense.com, http://ibisense.com

Mikko Syrjälahti

Ibisense, Sinimäentie 10 A, 02630 Espoo, Finland
Tel: +358 40 552 8136, mikko@ibisense.com, http://ibisense.com
OPEN SOURCE DATABASES, INDUSTRIAL AUTOMATION

ABSTRACT

Process automation systems existed for decades. Early implementations of such systems were built with help of mechanical sensors and actuators. Such early automation systems were producing rather small amounts of data, which was perhaps logged through chart recorders and manual processes. It is only with an appearance of computerized systems the industrial automation systems became way more complex and could now generate astonishing amounts of data which requires building scalable data storage facilities. In this regard, open-source solutions can serve as the bases for implementing such systems. In this paper we overview several open-source databases and present some experimental results.

1 INTRODUCTION

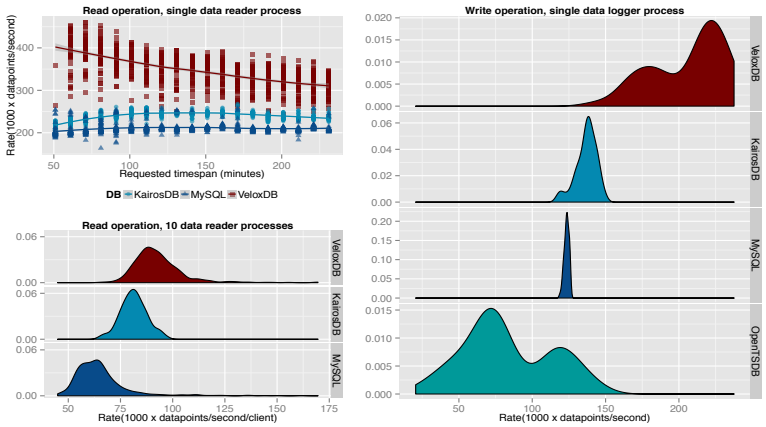
In the past few years, several open source projects were initiated with a goal of implementing a general-purpose time-series storage on top of open-source databases such as for example MySQL, Cassandra/2/, and Hbase/1/, MongoDB. The key advantage of any open-source projects is that their usage allows introducing new features easily. For example, support for new data types, custom data aggregation and filtering functions typically can be easily added without spending much time on implementing and supporting core features. Another advantage of any well-matured open-source projects is that the maintenance costs can be reduced since the code-base can be supported by third-party companies or even by the community. One of the most popular and matured relational databases today is MySQL. MySQL can be potentially used to store time-series data if for example the data set is of relatively small size. The problem can occur however if the amount of data for a single tag (by tag we assume the data source, sensor) becomes large. In the past few years' non-relational SQL (NoSQL) databases gained significant popularity and became attractive for storing time-series data. One of such databases is Apache Cassandra – fully distributed key-value store (master-master architecture in which each node has the same role as any other node in the system). There are separate projects, which provide API for storing time-series data in Cassandra. One such project is KairosDB. Similarly to Cassandra, HBase is another distributed NoSQL database. Although HBase, similarly to Cassandra, allows spreading the load across multiple nodes, HBase is harder to maintain and deploy. There are open-source projects that provide high-level API for storing time series data in HBase. One such project is OpenTSDB. MongoDB is another database that has received much attention in the past few years. In essence, MongoDB is a distributed document-oriented database. Although, MongoDB was originally designed for storing JSON-encoded documents, it can be used to store time-series data as well (For example, according to/3/ Siemens is using MongoDB for such purpose). Finally, we should mention

InfluxDB - a database, which can be easily integrated with other open source projects allowing creating a monitoring solution in a matter of hours.

2 EXPERIMENTAL RESULTS

We performed a series of measurements to evaluate several databases of our choice. Thus, we have measured the performance of MySQL (deployed on a single node), our own implementation of time-series database and KairosDB both being using 3 node Cassandra cluster and OpenTSDB (which was using HBase as the storage backend). We present these results in Figure 1.

Figure 1. *Read and write performance for different open-source databases. We measured write performance using a single data logger, which was constantly pushing 10^6 scalar datapoints to 10^3 different tags; to measure read operation from process historian, one or ten concurrent readers (depending on the setup) were fetching data for 200 randomly selected tags from a random time range (the requested timespan was, however, limited between about 1 to 3 hours).*



3 CONCLUSIONS

In this paper we have gave a high-level overview of several open-source projects. We believe that open-source NoSQL databases, such as Cassandra, can be suitable for building industrial grade data warehouse platforms for storing time-series data. Next, we have performed a series of experiments. Our experimental results demonstrate that NoSQL databases such as Cassandra can deliver good read and write performance (mainly because of the ability of scaling horizontally) when compared to traditional RDMS such as MySQL.

4 REFERENCES

D. Borthakur, J. Gray, J.S. Sarma, K. Muthukkaruppan, N. Spiegelberg, H. Kuang, K. Ranganathan, D.Molkov, A. Menon, S. Rash, R. Schmidt, and A. Aiyer. 2011. Apache Hadoop goes realtime at Facebook. In Proceedings of the 2011 ACM SIGMOD International Conference on Management of data (SIGMOD '11). ACM, New York, NY, USA, 1071-1080. DOI: 10.1145/1989323.1989438

A. Lakshman, P. Malik. Cassandra: a decentralized structured storage system. SIGOPS Oper. Syst. Rev. 44, 2 (April 2010), 35-40; DOI: 10.1145/1773912.1773922

Siemens. Smart City & Internet of Things http://www.mi.camcom.it/c/document_library/get_file?uuid=0442eeb7-2694-410d-a725-d51e610bccbe&groupId=10157