**Student Name: Nina Nguyen**

**Date: 10/5/2019**

**Topic: Database**

**Article:** [**AsterixDB: a scalable, open source BDMS**](https://dl-acm-org.proxy.seattleu.edu/citation.cfm?id=2733096)

**Citation: Alsubaiee, Sattam & Altowim, Yasser & Altwaijry, Hotham & Behm, Alexander & Borkar, Vinayak & Bu, Yingyi & Carey, Michael & Cetindil, Inci & Cheelangi, Madhusudan & Faraaz, Khurram & Gabrielova, Eugenia & Grover, Raman & Heilbron, Zachary & Kim, Young-Seok & Li, Chen & Li, Guangqiang & Ok, Ji & Onose, Nicola & Pirzadeh, Pouria & Westmann, Till. (2014). AsterixDB: A scalable, open source BDMS. Proceedings of the VLDB Endowment. 7.**

**Summary of Article:**

With all the information we are generating and collecting every day, where will we put it and how can we quick query it to analyze the data? This is what drives the development of AsterixDB, an open sourced full-function Big Data Management System, with features that set itself apart from other BDMS. AsterixDB was started in 2009 with an initial release in 2013 with continuing updates. AsterixDB is flexible NoSQL style data model with queries support, scalable runtime, partitioned LSM-based data storage, indexing, and many more. AsterixDB was then tested against famous Big Data systems such as Hive, Mongo, and System-X to test the performance. In the end, it shows that even in it’s early stages AsterixDB is highly competitive compared to more matured systems.

**Article Purpose:**

The purpose of the article was to share informational development of AsterixDB and how it is different than major Big Data Management System that is current available. Then a test was done to see performance of AsterixDB against other major Big Data Management System. The main keys that AsterixDB developer want their system to have are: semistructured data model for use cases ranging from “schema first” to “schema never”, full query language with at least the power of SQL, an efficient parallel query runtime, support for data management and indexing, support wide range of query sizes, support for continuous data ingestion, the ability to scale gracefully, and support for common Big Data datatypes. I think they did a great job with focusing the article on what make AsterixDB special by describing it’s functionality and features.

**Methodology:**

The authors started by describing the features of AsterixDB. For example, their instances can have an open Datatypes or closed Datatypes which can be declared by JSON. Unlike other BDMS, AsterixDB can read data externally or internally. Data feed is another interesting feature that was mentioned, it can ingests a continuous stream of data into a Dataset. AsterixDB also has it’s own data manipulation language called Asterix Query Language which was based off XQuery. To test the performance of AsterixDB, the authors used similar datasets to test against MongoDB, Apache Hive, and System-x (a commercial shared-nothing parallel relational DBMS). The queries used were read-only and insert operations from Record Lookup, Range Scan, Join, and Aggregation.

**Conclusion:**

AsterixDB is a full-function DBMS that is best put into a bucket that includes Big Data analytics, parallel RDBMS and NoSQL store. It takes features from all those systems and incorporated into AsterixDB. The authors mentioned that the purpose of the test was not to see is AsterixDB can outperform other systems, but to show that it can keep up with the big boys. And it did just that. Hive tends to fall behind because it doesn’t have direct support of indices so it has to do full scan of the datasets. Other than that, AsterixDB Schema and AsterixDB KeyOnly tend to have the same performance of System-X and MongoDB.

**Article Strengths:**

The article did an excellent job in breaking down the different features of AsterixDB in details. I thought it was very well formatted and neatly drove up to the tests and conclusion. I thought the methodology was well done too. The test was done on the same machine with same processing speed to make sure that doesn’t pollute the results. The authors also pointed out during the test that some system have limitations that will affect the test. For example, Hive has no direct support for index so it had to run a full scan.

**Article Weaknesses:**

There was a lack of future plans or implementations with AsterixDB. The authors mentioned that this is quite a new BDMS and there’s a lot of room for improvements, and they quickly mentioned a few features but didn’t really go into it. The article was from 2014, so a quick google showed that AsterixDB belongs to Apache now.

**Recommendation:**

The authors want to improve and add more functionality to AsterixDB in the future. Some of the things that they will work on is seamless integration with our Pregelix open source graph analytics system, potential use of HDFS for replicated LSM storage, and support for continuous queries and notifications to enable “declarative pub/sub” over Big Data.

**Checklist:**

Number of Authors: 23

Number of Citations of Article: 33

Number of Citations to other articles: 24

Methodology Explained (Yes/No): Yes

Technology Explained (Yes/No): Yes

Experiments and Data Reviewed (Yes/No): Yes

Conclusion Exist (Yes/No): Yes

Recommendations Exist (Yes/No): Yes