

# Non Relational Models for the Management of Large Amount of Astronomical Data

B. L. Martino<sup>1</sup>, M. Federici<sup>2</sup>

<sup>1</sup>*Associated INAF IAPS*

<sup>2</sup>*Istituto di Astrofisica e Planetologia Spaziali, INAF IAPS Via fosso del Cavaliere 100, 00133 Roma, Italy*

Corresponding author: brunolmartino@ias.i.cnr.it

## Abstract

The objective of this paper is the comparison between two different database typologies: the relational and the non-relational architecture, in the context of the applications related to the use and distribution of astronomical data. The specific context is focused to problems quite different from those related to administrative and managerial environments within which were developed the leading technologies on which are based the modern systems of massive storage of data. The data provided by astronomical instrumentation are usually filtered out by the front-end system (trigger, anticoincidence, DSP etc.), so they do not require special controls of congruence. Moreover, the related storage systems must be able to ensure an easy growth, minimizing human systemistic interventions and automating the related actions. The use of a non-relational architecture (NoSQL), offers great advantages during the insertion of informations within a data base, while the response speed of the queries is mainly tied to their type and complexity.

**Keywords:** DBMS - MongoDB - NoSQL - MySQL - GSC catalog.

## 1 Using DBMS

A careful planning of the use of a file system allows to store informations in a rational way but, whatever the criterion used to organize an archive based only on files, it is not possible to build a search system that can guarantee sufficient flexibility. The user of a database, conceived in this manner, is required to know in detail its structure and it's organization. Almost any application focused on the analysis of astronomical data may read data in FITS format (Wells et al., 1981) FITS format allows to add to a collection of data a set of additional information used to allow their better characterization. A DBMS (Data Base Management System) is a software infrastructure designed to operate on large data sets with the goal of optimizing the data:

- storage
- access
- sharing
- protection

A DBMS allows, through the use of its command language, the imposition of constraints of consistency, the creation of indexes to improve performance and retrieval of data independently from their physical representation. Well-known examples of the use of advanced databases in high-energy physics are represented

by SPIRES (Bourne et al., 2003), OPERA (Agafonova et al., 2009) and !CHAOS (Bisegni, 2012) and is consolidated practice their adoption as a storage medium in the ground segment.

### 1.1 Relational models

The relational databases today are the most popular; the model they are based it was proposed by Codd (1970). The relational model allows access to data at the logical level by providing a complete independence from their physical organization. The logical representation of the data in the relational model is based on the concept of relationship, in algebraic terms; it is common to use the term "table" in place of "algebraic relationship" and the term "relationship" to indicate an association between the data. The language SQL (Date et al., 1997) is the standard language "de facto" for defining, manipulating and interrogation of relational databases; this is a declarative language and not procedural. In the relational model a logical unit of work is defined transaction and is constituted by a sequence of operations of reading and writing, which must meet some properties, known as ACID property (from Atomicity, Consistency, Isolation, Durability). Atomicity: a transaction is an atomic unit of operations, validated or canceled depending on whether or not they reach a successful conclusion (rollback / commit); Consistency: at

the end of the transaction if the initial state is correct, even the final state must be; Isolation: the action of a transaction should not interfere with each other; Duration: the effects of a transaction must be persistent; The most popular relational databases are ORACLE (Kunh, 2010) and MySQL (Schwartz, 2012)).

## 1.2 Non relational models

Recently, have been developed a series of new DBMS systems, to provide an high horizontal scalability, in order to achieve high performance in the read / write operations of database distributed across multiple servers geographically delocalized (cloud). Many of these new systems are called NoSQL data stores. The definition of NoSQL, which stands for "Not Only SQL" or "Not Relational", was used for the first time in 1998 for an open source relational database. The non-relational systems, do not attempt to provide the classic ACID guarantees, typical of relational databases, but embrace the model B.A.S.E. (Basic Available, Soft-state, Eventual consistency) where some constraints are relaxed: Basic Available: the database can operate even if a part is no longer available; Soft-state: the system status may change over time even in absence of input data; Eventual consistency: the data may not be updated immediately, but will be consistent throughout the system within a finite time As stated by the CAP theorem (Lynch et al., 2002) can not be achieved Consistency, Availability and Partition tolerance at the same time but only two of these features at a time. The systems based on non-relational architecture follow the BASE model and are those that allow to overcome the major limitation of the RDBMS (Relational Data Base Management System): the scalability. In many areas, consistency and/or availability offered by relational databases are not essential, e.g. in astronomy and astrophysics applications. In a highly available and tolerant partitioning system, alteration of data base will reach all nodes not instantly but within a finite time, if a reading is done on a node is not synchronized with the last write, it returns the last valid data (stale data).

## 2 DBMS NoSQL

A property of NoSQL database is to be free of pattern (schemaless), with consequent advantages and disadvantages like ease of deployment, but sometimes, more difficulties to construct complex queries. The non-relational logical models not have the same expressive power of the relational model and can be classified into four main families: key-value, column-family, document store and graph. The most common are the MongoDB NoSQL architecture (Chodorow, 2011), HBase (George, 2011) and Cassandra (Lakshman et al., 2010); their characteristics are summarized in Figure 1. MongoDB

is a document-oriented database, which is based on aggregates, which may have a structure with multiple hierarchical levels and groups that can be variously indexed. MongoDB uses the JavaScript language, which is inherently single-threaded while Cassandra uses the language CQL (Cassandra Query Language, simplified variant of SQL), which allows the management of distributed databases. Both rely on the file system of the machine on which they were installed without introducing data abstractions. HBase is a distributed database (based on the project Hadoop (Sammer, 2012) written in Java, that uses storage devices located on different hosts also geographically distributed, interconnected via networks. HBase uses HDFS, a portable and scalable distributed file system initially developed for the framework Hadoop.

	MongoDB	Hbase	Cassandra
Architecture	master/slave	master/slave	p2p
File system	local	HDFS	local
Parallel progr. mod.	map/reduce	map/reduce	map/reduce
Consistency	strong	strong	weak

**Figure 1:** Key features of NoSQL architectures.

To carry out complex operations is used the MapReduce paradigm (Dean et al., 2004), which allows to divide the computation in many elaborations of lower complexity to achieve the processing of large datasets in parallel on multiple cores/CPU/computer.

## 3 MySQL vs MongoDB

In order to evaluate the possible use with advantage of non-relational architectures in astronomy and astrophysics area we realized the MDBirs system (showed in Figure 2), composed by 10 PC, equipped with Intel i7, 16Gbytes of RAM, 2 HD 1Tbytes connected via ethernet LAN at 1Gbit of speed



**Figure 2:** The MDBirs System.

In order to assess objectively the behavior of the system tests were carried out so as not to make use of the advanced characteristics of the query languages used. For this reason it was chosen as the working set of data the GSC catalog (Stars General Catalog). GSC is composed of a single table and is used primarily to provide support for the planning and guiding stars of the HST observations (Dalcanton, 2009), the JWST (Gardner et al., 2006), Gaia (Busso et al., 2012) and some ground-based telescopes of new generation. Furthermore, the services provided by the machines on which they are installed databases are not affected by processes (user or system) not strictly essential to their functioning and the software installed on the machines which host the test has been aligned to the same versions. The databases under test are the two most widely used open source architectures in the world, MySQL for its versatility, speed and diffusion and MongoDB for its attitude to the horizontal growth (scalability) and its robustness. Starting from the data available, have been generated some of the series of samples of increasing amplitude, so as to highlight the behavior of the two systems both with regard to populate the database, than to retrieve the information of interest. The test was performed using queries significant from the point of view of the community of users, which allow to highlight the behavior of databases in real operating conditions. In particular:

- query1: selection on HEALPix (mapping system applied to the celestial sphere)
- query 2: selection of objects in a spatial region which satisfy some conditions in magnitude and color
- query 3: selection of non-stellar objects which meet a condition in color
- query 4: computing of the average magnitude, resolution of one degree square

## 4 Test Results

The database engine installed by default in MySQL since version 5.0 (InnoDB) is transactional, ACID compliant and uses the row-level locking strategy (constraint of exclusive use for the time necessary to perform the required actions). The data entry test was made from data provided in CSV (text data whose fields are marked by the separator character comma.) As shown by the graph in Figure 3, the behavior of MongoDB is faster than MySQL, thanks to the absence of checks carried out on the integrity of the data; information are stored in files as key-value pairs.

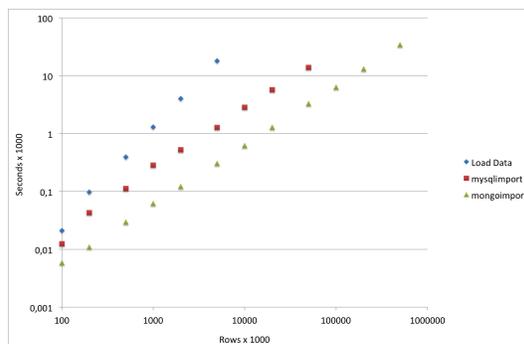


Figure 3: Insert performances vs data set size.

Regarding the execution time of the test queries, the results are shown in the graph of Figure 4 and are substantially comparable. In the case of the GSC catalog, the occupation of the relevant MySQL table is about 116 Gbytes of data collection while the corresponding MongoDB occupies about 657 Gbytes (the size is 5 times bigger). The tables were not associated with indexes, so searches are performed on the whole set of data with a time dependent to amplitude of the set itself.

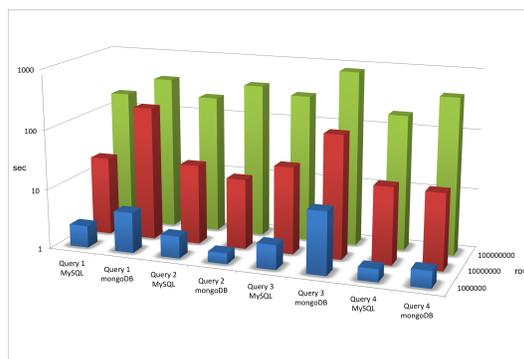


Figure 4: Queries performances vs data set size.

## 5 Future Activities

MongoDB is a good choice to obtain good safety and performances using local data sharding. Our goal is to achieve a full geographic data delocalization (CLOUD) in order to obtain: continuity of access to data (total disaster recovery) to the user community optimized average access time We evaluated two possible scenarios: HBase: Hadoop Based NoSQL Data Base Cassandra: The Apache Cassandra database HBase is a master-slave system including two types of machines: HMaster: access control HRegionserver: local data replication By using the Stargate plugin can be achieved local caches able to speed up response times by minimizing the network traffic to the master. Every region server keeps a copy of the data so the system ensures very

high reliability. It is a NoSQL solution initially developed by Facebook which has in the P2P architecture its focal point.

## 6 Conclusions

MongoDB has proven much more efficient and fast in data entry and is particularly suited to the management of flows of data to be stored without downtime. In the configuration we used the rate of loading is about 15000 rows/sec (the average rows length is 256 bytes). By using more powerful hardware such as high speed network interfaces (10 Gbit/sec or higher) it's possible to greatly increase the speed of data acquisition. Regarding the search of data in the database the results show a behavior strongly dependent on the number of records and the presence or absence of critical elements such as complex calculations. The command language of MongoDB does not contain any advanced mathematical primitives as in the case of MySQL, which can rely on a large library of mathematical functions. The use of MongoDB on structures designed to be handled by a RDBMS is inappropriate because it does not exploits the potential of its schemaless organization (nosql).

## Acknowledgement

We thanks Paola Marrese by ASDC-ASI for providing us the test data, , the Director of IAPS-INAF Prof Pietro Ubertini, and a special thanks to Franco Giovannelli and his staff for allowing us to participate to this workshop

## References

- [1] Wells, D. C., Greisen, E. W., and Harten, R. H.: 1981, A Flexible Image Transport System. *Astronomy & Astrophysics Supplement Series*, 44, 363-370.
- [2] Bourne, C. P., Hahn, T. B.: 2003, A history of online information services, 1963-1976, MIT press.
- [3] Agafonova, N., et al.: 2009, The detection of neutrino interactions in the emulsion/lead target of the OPERA experiment, *Journal of Instrumentation*. doi:10.1088/1748-0221/4/06/P06020
- [4] Bisegni, C.: 2012, Thesis in Computer Science, University Tor Vergata, Rome.
- [5] Codd, E.F.: 1970, A relational model of data for large shared data banks, *Communications of the ACM* 13 (377-387). doi:10.1145/362384.362685
- [6] C. J. Date, C. J., Darwen, H.: 1997, A Guide to the SQL standard: a users guide to the standard database language SQL, Addison Wesley.
- [7] Kuhn, D.: 2010, Pro Oracle Database 11g Administration, Apress. doi:10.1007/978-1-4302-2971-1
- [8] Schwartz, B.: 2012, High Performance MySQL: Optimization, Backups, and Replication, O'Reilly.
- [9] Lynch, N., Gilbert, S.: 2002, Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services, *ACM SIGACT News*, Volume 33 Issue 2, pg. 51-59.
- [10] Chodorow, K.: 2011, Scaling MongoDB, O'Reilly.
- [11] George, L.: 2011, HBase: The Definitive Guide, O'Reilly.
- [12] Lakshman, A., Malik, P.: 2010, Cassandra: a decentralized structured storage system, *ACM SIGOPS Operating Systems Review*, vol. 44, no.2. doi:10.1145/1773912.1773922
- [13] Sammer, E.: 2012, Hadoop Operations, O'Reilly.
- [14] Dean, J., Ghemawat, S.: 2004, MapReduce: Simplified Data Processing on Large Clusters, in *OSDI'04: Sixth Symposium on Operating System Design and Implementation*, San Francisco, CA, December.
- [15] Dalcanton, J.J.: 2009, 18 years of science with the Hubble Space Telescope, *Nature* Jan 1;457(7225):41-50.
- [16] Gardner, J. P., et al.: 2006, The James Webb Space Telescope, *Space Science Reviews*. doi:10.1007/s11214-006-8315-7
- [17] Busso, G., De Angeli, F., Montegriffo, P.: 2012, The GAIA photometric data processing, in *Proc. SPIE 8442, Space Telescopes and Instrumentation*