A Survey Of Workload Management In Big Data Storage

Article *in* Journal of Advanced Research in Dynamical and Control Systems · October 2020

Dr. Manoharan Subramanian
Ambo University

A Survey Of Workload Management In Big Data Storage

Velmurugan Lingamuthu,Professor, Computer Science, Institute of Technology, Ambo University, Ethiopia
Sasikumar Perumal, Assistant Professor, Information Technology, Institute of Technology, Ambo University, Ethiopia
Manoharan Subramanian, Assistant Professor, Computer Science, Institute of Technology, Ambo University, Ethiopia

Abstract-The information explosion has made the world of data over the last five years. The advancement of mobile technology, the availability of tablets and smartphones, and the rapid growth of social media have all contributed to both production and consumption of data at never-before-seen volumes. Other contributing factors have been recommendation engines, cool new visualization capabilities for business intelligence (BI), advances in software. Different types of data with varying degrees of complexity are produced at multiple levels of velocity. A huge increase in data storage and processing requirements has led to Big Data, for which next generation storage systems are being designed and implemented. As Big Data stresses the storage layer in new ways, a better understanding of these workloads and the availability of flexible workload generators are increasingly important to facilitate the proper design and performance tuning of storage subsystems like data replication, metadata management, and caching. This paper is a survey paper, which reveals the workload management in Big Data storage.

Keywords---mobile technology, social media, business intelligence, Big Data.

Introduction

A workload can be defined as the execution and completion of a task that utilizes a mix of resources—for example, processing power, storage, disk I/O, and network bandwidth. At any given time, any system that is processing information is executing a workload. Processing of workloads is common to the world of data warehousing and BI, and very applicable to the underlying information architecture. Big data benchmark suites must include a diversity of data and workloads to be useful in fairly evaluating big data systems and architectures. However, using truly comprehensive benchmarks poses great challenges for the architecture community. First, we need to thoroughly understand the behaviours of a variety of workloads. Second, our usual simulation-based research methods become prohibitively expensive for big data. As big data is an emerging field, more and more software stacks are being proposed to facilitate the development of big data applications, which aggravates these challenges. No Big Data storage system metadata trace is publicly available and existing ones are a poor replacement. The practice of analytics involves applying science and computing technology to vast amounts of raw data to yield valuable insights, and the "analytics" label covers a wide array of applications, tools, and techniques. While there are many analytic variants and subspecialties—predictive analytics, in-database analytics, advanced analytics, web analytics, and so on—this text focuses on the characteristic demands that nearly all analytic processing problems place on modern information systems. It is referred to these demands as an analytic workload. Every data processing problem has its own unique workload, but analytic workloads tend to share a set of attributes, with strong design and deployment implications for the processing systems assigned to handle these workloads.

Analytic vs. Transactional work load

Transactional processing is characterized by a large number of short, discrete, atomic transactions. The emphasis of online transaction processing (OLTP) systems is (a) high throughput (transactions per second), and (b) maintaining data integrity in multi-user environments. Analytics processing is characterized by fewer users (business analysts rather than customers and POS operators) submitting fewer requests, but queries can be very complex and resource intensive. Response time is frequently measured in tens to hundreds of seconds. Transactional and analytics processing tasks constitute very different workloads, and transactional and analytic information systems are designed with these differences in mind.

Analytic Workload

Before selecting, constructing, or deploying an analytic infrastructure, it makes sense to try to understand the basic characteristics and requirements of an analytic workload. Later, in addition to helping us outline an effective analytic infrastructure, these workload criteria can be used to evaluate a specific project or problem, yielding a rough

**Corresponding Author: Velmurugan Lingamuthu, Email : vel.erode@gmail.com.*

Article History: Received: April 04, 2019, Revised: September 25, 2019, Accepted: October 28, 2019

measure of analytic complexity. An analytic workload will exhibit one or more of the following characteristics, each of which elevates a given workload's degree of difficulty:

- Extreme data volume
- Data model complexity
- Variable and unpredictable traversal paths, patterns, and frequencies
- Set-oriented processing and bulk operations
- Multi-step, multi-touch analysis algorithms
- Complex computation
- Temporary or intermediate staging of data
- Change isolation/data stability implications

Rating each of these characteristics on a 1-10 scale yields the following general comparison for some idealized sample workloads:

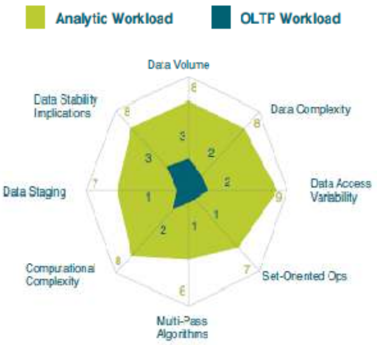


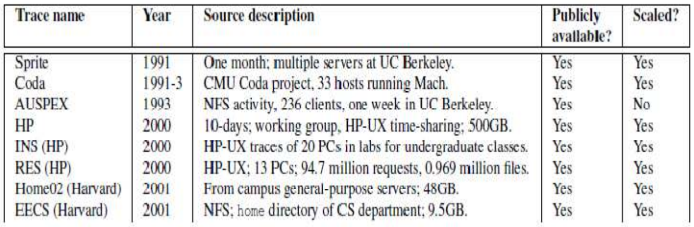
Figure 1. Analytic and transactional Workloads

Mapreduce Clusters

MapReduce clusters offer a distributed computing platform suitable for data-intensive applications. MapReduce was originally proposed by Google and its most widely deployed implementation, Hadoop, is used by many companies including Facebook, Yahoo and Twitter. MapReduce uses a divide-and-conquer approach in which input data are divided into fixed size units processed independently and in parallel by map tasks, which are executed distributed across the nodes in the cluster. After the map tasks are executed, their output is shuffled, sorted and then processed in parallel by one or more reduce tasks.To avoid the network bottlenecks due to moving data into and out of the compute nodes, a distributed file system typically co-exists with the compute nodes for Google's MapReduce and HDFS for Hadoop. MapReduce clusters have a master-slave design for the compute and storage systems. In the storage system, the master node handles the metadata operations, while the slaves handle the read/writes initiated by clients. Files are divided into fixed-sized blocks, each stored at a different data node. Files are read-only, but appends may be performed in some implementations. For the sake of simplicity, in this proposal we refer to the components of the distributed file system using the HDFS terminology, where name node refers to the master node and data node refers to the slave.These storage systems may use replication for reliability and load balancing purposes. GFS and HDFS support a configurable number of replicas per file; by default, each block of a file is replicated three times. HDFS's default replica placement is as follows. The first replica of a block goes to the node writing the data; the second, to a random node in the same rack; and the last, to a random node. This design provides a good balance between being insensitive to correlated failures (e.g., whole rack failure) and minimizing the inter-rack data transmission. A block is read from the closest node: node local, or rack local, or remote.

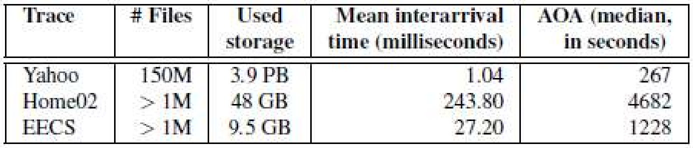
Towards A Metadata Workload Model

To support the arguments through an analysis of: a Big Data namespace metadata trace from Yahoo and two traces used in prior work (Home02 and EECS in Tables 3.1-3.2; which are the most recent public traces used in the papers surveyed. The Big Data trace comes from the largest Hadoop cluster at Yahoo (4000+ nodes, HDFS); this is a production cluster running data-intensive jobs like processing advertisement targeting information. The namespace metadata trace has a snapshot of the namespace (04/30/2011) obtained with Hadoop's Offline Image Viewer tool, and a 1-month trace of namespace events (05/2011) obtained by parsing the name node (MDS) audit logs. See the tables for workload details.



Trace name	Year	Source description	Publicly available?	Scaled?
Sprite	1991	One month; multiple servers at UC Berkeley.	Yes	Yes
Coda	1991-3	CMU Coda project, 33 hosts running Mach.	Yes	Yes
AUSPEX	1993	NFS activity, 236 clients, one week in UC Berkeley.	Yes	No
HP	2000	10-days; working group, HP-UX time-sharing; 500GB.	Yes	Yes
INS (HP)	2000	HP-UX traces of 20 PCs in labs for undergraduate classes.	Yes	Yes
RES (HP)	2000	HP-UX; 13 PCs; 94.7 million requests, 0.969 million files.	Yes	Yes
Home02 (Harvard)	2001	From campus general-purpose servers; 48GB.	Yes	Yes
EECS (Harvard)	2001	NFS; home directory of CS department; 9.5GB.	Yes	Yes

Table 1: Description of the traces used by one or more of the surveyed papers; references in [3].



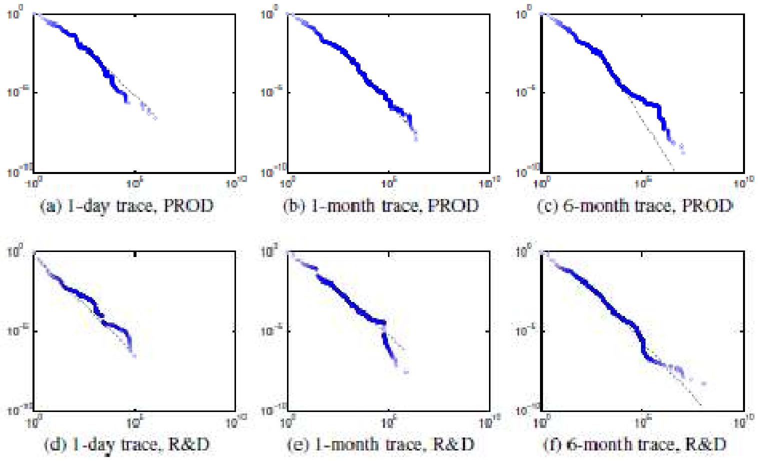
Trace	# Files	Used storage	Mean interarrival time (milliseconds)	AOA (median, in seconds)
Yahoo	150M	3.9 PB	1.04	267
Home02	> 1M	48 GB	243.80	4682
EECS	> 1M	9.5 GB	27.20	1228

Table 2: Traces analyzed in this Chapter; AOA: age at time of access. # files includes files created or deleted during the trace.Storage system workloads are multi-dimensional and can be defined by several characteristics like namespace size and shape, arrival patterns, and temporal locality patterns. One of these dimensions is the temporal locality present in the workload, which is to measure through the distribution of the age of a file at the time it is accessed (AOA). For every operation (namespace metadata event), it is calculated how old the file is and use this information to build a cumulative distribution function (CDF) that represents the workload in this dimension. This dimension is chosen because it is one that is very relevant to namespace metadata management, since the temporal locality of the workload has an incidence in mechanisms like load balancing, dynamic namespace partitioning/distribution, and caching.

Analysis of Two Mapreduce Workloads

Other characteristics of the workloads, not directly related to the access patterns, are also presented to help provide a broader characterization of the workloads and which may be of interest to other researchers.

File popularity: Figure 2 shows the Complementary Cumulative Distribution Function (CCDF) of the file accesses (opens), for both clusters, for different periods: first day of the trace, first month of the trace and full six-month trace. The CCDF shows P(X ¸ x), or the cumulative proportion of files accessed x or more times. The dashed line shows the best Power Law fit for the tail of the distribution. Files not accessed during the trace were ignored for these plots;Figure 2: Complementary Cumulative Distribution Function (CCDF) of the frequency of file accesses (opens), for increasingly larger traces.



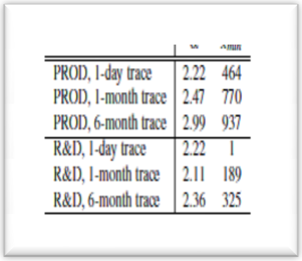
(a) 1-day trace, PROD (b) 1-month trace, PROD (c) 6-month trace, PROD

(d) 1-day trace, R&D (e) 1-month trace, R&D (f) 6-month trace, R&D

The CCDF shows P(X ¸ x), or the cumulative proportion of files accessed x or more times in the trace. The dashed line shows the best Power Law fit for the tail.

*Corresponding Author: Velmurugan Lingamuthu, Email : vel.erode@gmail.com.

Article History: Received: April 04, 2019, Revised: September 25, 2019, Accepted: October 28, 2019



	α	xmin
PROD, 1-day trace	2.22	464
PROD, 1-month trace	2.47	770
PROD, 6-month trace	2.99	937
R&D, 1-day trace	2.22	1
R&D, 1-month trace	2.11	189
R&D, 6-month trace	2.36	325

Table 3: Best fit of file access frequency (Figure 2) to a Power Law. a: scaling parameter, xmin: lower bound of power-law behavior.a xmin.

Since file access patterns in other workloads exhibit Power Law behavior (or Zipf Law if ranked data is analyzed), we provide the results of the best fit of the tail of the distribution to a Power Law. To find the best fit, we apply the methodology (and toolset) . Results are shown in Figure 2 and Table 3. The latter shows the Power Law scaling parameter (a) and xmin, the value where the fitted tail begins. xmin is chosen so that the Kolmogorov-Smirnov goodness-of-fit test statistic (D)—which is the maximum difference between the two CDF curves—is minimized.

Conclusions and Future Directions

In this paper, the problem of workload characterization and modelling for Big Data storage systems is studied. Specifically, studied how MapReduce interacts with the storage layer and created models suitable for the evaluation of namespace metadata management subsystems. To deepen our understanding of these emerging workloads, it is analysed the two 6-month namespace metadata traces from two large clusters at Yahoo. Our workload characterization provided good insight on how to properly design and tune systems for MapReduce workloads. To the best of our knowledge, this is the only comprehensive study of how MapReduce interacts with the storage layer.

References

[1] Tachyon. tachyon-project.org, March 2013. Last accessed: April 18, 2013.

[2] Cristina L. Abad, Yi Lu, and Roy H. Campbell. DARE: Adaptive data replication for efficient cluster scheduling. In Proceedings of the International Conference on Cluster Computing (CLUSTER), 2011.

[3] Cristina L. Abad, Huong Luu, Yi Lu, and Roy H. Campbell. Metadata workloads for testing Big storage systems. Technical report, UIUC, 2012.hdl.handle.net/2142/30013.

[4] Cristina L. Abad, Huong Luu, Nathan Roberts, Kihwal Lee, Yi Lu, and Roy H. Campbell. Metadata traces and workload models for evaluating Big storage systems. In Proceedings of the IEEE/ACM Utility and Cloud Computing Conference (UCC), 2012.

[5] K, m. S. K., alias, . M. & r., . S. K. (2018) a review on novel uses of vitamin e. Journal of Critical Reviews, 5 (2), 10-14. doi:10.22159/jcr.2018v5i2.24282

[6] Vo, Nam Xuan, Trung Quang Vo, Ha Thi Song Nguyen, and Thuy Van Ha. "The Economic Evaluation in Vaccines - A Systematic Review in Vietnam Situation." Systematic Reviews in Pharmacy 9.1 (2018), 1-5. Print. doi:10.5530/srp.2018.1.1

[7] Kumar, N. Suresh, M. Arun, and Mukesh Kumar Dangi. "Remote sensing image retrieval using object-based, semantic classifier techniques." International Journal of Information and Communication Technology 13, no. 1 (2018): 68-82

[8] .Nitin Agrawal, William Bolosky, John Douceur, and Jacob Lorch. A five year study of file-system metadata. ACM Transactions on Storage (TOS), 3, 2007.

[9] Sadaf Alam, Hussein El-Harake, Kristopher Howard, Neil Stringfellow, and Fabio Verzelloni. Parallel I/O and the metadata wall. In Proceedings of the Petascale Data Storage Workshop (PDSW), 2011.

[10] "Understanding Analytic Workloads", Meeting the complex processing demands of advanced analytics, IBM.

[11] "Avoid the Workload Bottleneck" by Krish Krishnan, Isolating workloads helps revolutionize data warehouse design by capitalizing on recent advances in technology, September 20, 2013.