Cross-Layer SLA Management for Cloud-hosted Big Data Analytics Applications

Xuezhi Zeng[1]

Supervision Panel: [1]Rajiv Ranjan, [1]Peter Strazdins, [2]Saurabh Kumar Garg, and [3]Lizhe Wang

[1]Australian National University, Australia
{xuezhi.zeng, rajiv.ranjan, peter.strazdins}@anu.edu.au

[2]University of Tasmania, Australia
Saurabh.Garg@utas.edu.au

[3]Chinese Academy of Sciences, China
lzwang@ceode.ac.cn

***Abstract*—As we come to terms with various big data challenges, one vital issue remains largely untouched. That is service level agreement (SLA) management to deliver strong Quality of Service (QoS) guarantees for big data analytics applications (BDAA) sharing the same underlying infrastructure, for example, a public cloud platform. Although SLA and QoS are not new concepts as they originated much before the cloud computing and big data era, its importance is amplified and complexity is aggravated by the emergence of *time-sensitive* BDAAs such as social network-based stock recommendation and environmental monitoring. These applications require strong QoS guarantees and dependability from the underlying cloud computing platform to accommodate *real-time responses* while handling ever-increasing complexities and uncertainties. Hence, the over-reaching goal of this PhD research is to develop novel simulation, modeling and benchmarking tools and techniques that can aid researchers and practitioners in studying the impact of uncertainties (contention, failures, anomalies, etc.) on the final SLA and QoS of a cloud-hosted BDAA.**

Keywords- Cloud Computing, Big Data, Service Level Agreement

I. INTRODUCTION

Two fastest-moving technologies are identified in Gartner Inc.'s 2012 Hype Cycle for Emerging Technologies and are further listed among the technologies that represent the Digital Marketing stage on the Gartner Inc.'s 2014 Hype Cycle. First is the Cloud Computing [1] that enables organization to access applications, resources, and services over the Internet on-demand basis via a "usage-based" business model. The second disruptive technology is the Big Data, which is defined as the practice of collecting and analyzing structured and unstructured data sets flowing at a volume and velocity that is too large and too fast to store, process, and interpret manually or using traditionally data management applications.

Cloud computing and big data are that two technologies that are inherently and increasingly getting entwined with each other. It is well understood that cloud computing platforms are well suited for hosting BDAA as they offer an elastic hardware resources (e.g. CPU, Storage, and Network) that can be scaled on-demand for handling uncertain data volume, variety, velocity, and query types [2, 3]. However, the key complications in hosting BDDA on clouds arise from requirement of generation real-time responses. For instance, a lot of papers have consider on personalized stock recommendations [3] which use live Twitter sentiments and stock data to analyze the stock market movement and correlation in order to recommend buying and selling points and promising combinations of stocks to investors in real time. In this time-sensitive application, data analysis results should be fed back to the investors through highly efficient processing of queries, so that investors can decide just-in-time when and which stocks they should purchase or sell. Such time-sensitive applications cannot tolerate delays in the data processing (the fast movement of stock market, the impact of politics, economy and many other factors) required for real-time decision making. Therefore, data analysis delays could lead to significant monetary losses.

To effectively support time-sensitive BDDA, new solutions are required that can ensure the QoS guarantees and dependability. Such requirements can be enforced and expressed via SLAs. SLA can be defined as the formal contract between customers and providers that states the nature and the scope of QoS metrics relevant to a provided service. For instance, many companies who offer resource- or infrastructure-level SLAs (iSLAs) (e.g., Amazon EC2 promises 99.95% availability for its CPU, Storage, and Network resources), only provide cloud services in the "best-effort" way, few of them deliver strong guarantees in terms of QoS of BDDA (e.g., minimize relevant stock recommendation delay)[3].

To illustrate this real-time SLA management problem further let us consider the following BDAA platform (Fig. 1) hosted on public clouds and architected based on the Lambda Architecture [4]. The platform consists of the following: Data ingestion frameworks (e.g. Apache Kafka, and Amazon Kinesis enable high-throughput and low-latency queuing of real-time messages; Parallel programming frameworks (e.g. Apache Hadoop and Apache Storm) support the development of applications for processing historical and streaming data across parallel clusters of cloud resources; Data storage frameworks (e.g. MongoDB, BigTable, MySQL, and Cassandra) aid in management of structured, unstructured, and semi-structured data; Cloud Computing platforms (e.g. high performance and elastic datacentre cloud resources) provide on-demand access to pay-as-you-go hardware resources (e.g. CPU, GPUs, storage, and network).

In Fig. 1, there are three layers, namely SaaS PaaS and IaaS which is a well-understood architectural pattern for a cloud hosted application stack. For time-sensitive BDAAs the hard challenge is to guarantee SaaS-level aSLAs (e.g. minimize event detection delay, maximize application availability) by carefully managing PaaS-level pSLAs (e.g. minimize stream processing latency, minimize batch processing response time) and IaaS-level iSLAs (e.g.

minimize network throughput and latency, maximize CPU Utilization). It is clear that SLAs (i.e. aSLAs, pSLAs, and iSLAs) and QoS metrics varies considerably across the layers; hence developing techniques to guarantee these layered SLAs remains an open and hard problem due to the complexities discussed in Section II.

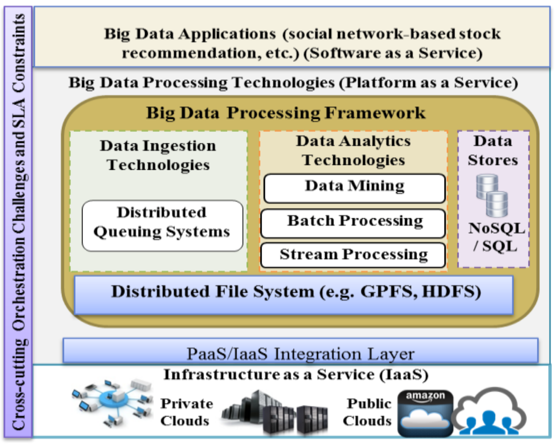

Figure 1. BDDA platform consisting of multiple data processing frameworks hosted over shared cloud resources

The rest of this paper is organized as follows: Section II describes the key research problems in detail. Section III explains the proposed approach and methodology towards to the research problems in Section II. The related work is briefly presented in Section IV. In Section V and VI expected contribution and preliminary work are discussed.

II. RESEARCH PROBLEMS

After conducting thorough literature review, I identified the following hard research problems that will be investigated in my PhD research.

***Q1*: How to specify and reason multi-layered aSLAs, pSLAs, and iSLAs**

Identifying and defining multi-layered SLAs and relevant QoS metrics for cloud-hosted BDDAs is a difficult problem due to the flowing weaknesses of the existing approaches:

- **Inconsistency**: If we examine current SLAs for cloud services in the market, they are typically published on cloud service providers' websites in various formats, non-standardized templates and terms which varies greatly with different cloud service providers. For example, Amazon refers *availability* as Monthly Uptime Percentage, while GoGrid refers to the server uptime and Rackspace use monthly availability.
- **Incompleteness**: Current published SLAs are all predefined emphasizing only availability and ignoring other important BDDA related QoS metrics and resulting multi-layered aSLAs, pSLAs, and iSLAs. Despite the rapid emergence of cloud computing and BDDA, little work has been done in this area. For instance, none of the existing data ingestion, data storage, parallel and distributed programming frameworks, and scalable data mining frameworks guarantee aSLA satisfaction.

This leads to the following research questions: How should we define and enforce the cross-layered SLAa (aSLA, pSLA and iSLA)? How the various QoS metrics relevant to PaaS (big data processing frameworks) and Iaas (hardware resources) layers should be combined in order to give a holistic view about the relevant pSLAs and iSLAs? Can we develop a new holistic SLA framework for the BDDA platform to regulate the behavior between analytics service customers and providers? Can we set a unified template of SLA specification and reasoning to guide the interaction BDDA platform providers and customers in the future?

***Q2*: How to develop an approach to study and analyze the SLA-based deployment BDDAs under multiple types of uncertainties**

To support SLA-based deployment of multiple and heterogeneous big data processing frameworks, we must consider which analytics workloads should be combined on a common physical server, to minimize resource contention and interference. Although there are many technologies aim at providing promising platforms to address the challenges posed by Big Data, it is extremely hard to design, develop, and implement an contention-free BDDA platform[3, 5]

This raises a set of research questions: Can we develop an approach that can help academics and practitioners in analyzing the impact of resource contention on aSLAs, pSLAs, and iSLAs due to deployment of multiple big data frameworks over shared hardware resources of public and private clouds in a repeatable and controllable manner. Conducting such a study in a real computing environment can be a challenge for several reasons:

- It is not cost-effective to procure or rent a large scale datacenter resource pool that will accurately reflect realistic application deployment and let practitioners experiment with dynamic hardware resource and big data processing framework configurations, and changing data volume, velocity, and variety.
- Frequently changing experiment configurations in a large-scale real test bed involves lot of manual configuration, making the performance analysis itself time-consuming. As a result, the reproduction of results becomes extremely difficult
- It is extremely hard to incorporate and control different types of failure behaviors and benchmarks across heterogeneous software and hardware resource - types in a real test bed environment.

The ultimate aim of my PhD research is to investigate approaches, techniques and tools to solve the above problems such that cross-layer SLA management for QoS guarantees can be assured to cloud-hosted BDDAs.

III. PROPOSED APPROACH AND METHODOLOGY

To solve *Q1*, I will focus on extensive review of the literature related to the topics of cloud computing and big data applications. Based on this, firstly, I will design an

SLA framework that can accommodate the requirements of BDDA platform. Then, I will define SLAs and QoS metrics at application), platform, and Infrastructure layers and specify the interrelationship among of them by investigating techniques such as Petri Nets and Reliability Block Diagrams. In addition, the existing SLA management approaches in cloud computing and big data environment will be analyzed and compared. We currently know that some QoS metrics are unique in big data environment compared with cloud computing environment. For instance, throughput and latency in distributed messaging system, response time in batch processing platform and for queries in transactional systems are unique QoS metrics.

To solve *Q2*, I will explore a novel and extensible SLA-oriented BDDA simulation and benchmarking platform by significantly extending the CloudSim[6] framework that can support modeling of heterogeneous data programming abstractions, heterogeneous data flows multi-workload processing, and hardware resource configurations. This will facilitate development of consolidation techniques and baseline workload models for deciding which and how big data workloads should be deployed on a shared infrastructure while minimizing contention and interference, such that we could easily conduct what-if analysis to understand the aforementioned resource contention and interference. The extended CloudSim will then support SLA evaluation templates that incorporate details on aSLA constraints, fault injection models, workload contention models, big data processing benchmarks, and configuration. The core novelty of extended CloudSim will be an extensible environment in which multiple big data processing frameworks and hardware resources can be simulated at the same time. To the best of our knowledge, there has been no investigation of prediction models that take into account impact of other events (hardware or software failures, contention) on performance metrics and SLAs. The newly extended CloudSim will support such performance evaluation.

IV. STATE OF THE ART

As regards to *Q1*, Chhetri et al. [7] proposed the automation of SLA establishment based on a classification of cloud resources in different categories with different costs, e.g. on demand instances, reserved instances and spot instances in Amazon EC2 cloud. The SLA management framework developed by Hsien [8] focuses on Software as a Service model of delivery in cloud computing only. Further details are provided on how the services can be integrated to support the concept of stability of the cloud community, especially for SaaS. Garg et.al [9] proposed a novel framework and a mechanism that measure the quality and prioritize Cloud services based on their ability to meet the user's QoS requirements. None of the above approaches consider *definition and specification* of the complex QoS metrics that fit requirements of BDDA platform. Further, they did not consider the cross-layered aSLAs, pSLAs, and iSLAs dependencies (see Fig. 1).

To the best of our knowledge, very few of the existing approaches address the problem of designing the cross-layer SLA management techniques and tools for managing BDDA platform. We could only find fairly limited threads on this topic. Authors in [10] proposed a SLA and cost-aware resource provisioning and task scheduling approach tailored for big data applications in the cloud. They proposed and compared cost-aware and SLA-based algorithms which provision cloud resources and schedule analytics tasks. However, they only considered the budget and deadline as the aSLAs and restricted their research to the batch (Apache Hadoop) layer. Nita et al. [11] discussed data transfers in the cloud, which affects performance in the case of Virtual Machine migration and of user submitted big data transfers. They described a method for big-data transfer optimization based on network characteristics and considered the constraints introduced by iSLA. However, this research mainly targets to minimize simplistic data transfer iSLA constraint.

As regards to *Q2*, recent work has investigated the impact of workload interference on the performance SLAs of hosted applications. New hardware design techniques change processor cache architecture partitioning or integrate novel insertion policies to pseudo-partition caches to reduce interference. Hardware-based approaches add complexity to processor architecture and are difficult to manage overt time. Govindan et al. [12] developed a scheme to quantify the effects of cache interference between consolidated workloads. However, the aforementioned techniques focus on interference issues of only one hardware resource type while ignoring others. Nathuji et al. [13] presented a control theory-based approach to consolidation that mitigates the effects of cache, memory, and hardware pre-fetching interference of co-existing workloads. However they focus on only CPU-bound or compute-intensive applications. Zhu etc.al [14] apply a Kalman Filter to model the interference caused by deploying contentious web applications on CP resources. It is worth noting that big data orchestration platforms such as YARN, Mesos and Amazon EMR have no support for detecting and handling resource contentions.

Over the last decades, many simulators were developed to facilitate research on various aspects of cloud computing infrastructures. They include NS2, CloudSim, iCanCloud and MR-CloudSim etc. While these simulators were widely adopted in the study of the behaviors of cloud computing applications, they unfortunately do not support modeling and simulation of diverse big data processing frameworks and cannot offer a QoS service.

Also, several benchmarks have emerged in the recent past including BigDataBench[15], BigBench, Hibench, PigMix, CloudSuite, and GridMix fueled by the needs of analysing the performance of different big data processing frameworks. These benchmark suites model workloads for stress testing one or more categories of big data processing frameworks. Among these frameworks, BigDataBench is a most comprehensive one as it constitutes workload models for NoSQL, DBMS, SPEs and batch processing frameworks. Primarily, BigDataBench targets the following application domains: search engine, social network, and e-commerce.

As a conclusion of the above related works targeted to the two research problems identified in Section II, to the best of our knowledge, there are neither concrete and practical SLA and QoS management framework tailored for BDDA platform. Further there is clear lack of software-based technique that can analyze interference among deployed big data workloads via simulation and benchmarking to understand how final aSLAs will be affected due to contention at PaaS and IaaS layers.

V. EXPECTED CONTRIBUTIONS AND IMPACT

The outcomes of this research will make significant scientific advances in understanding the theoretical and practical problems of SLA management BDDA platform.

The first contribution of this PhD thesis as regard to the *Q1* include i) a customized conceptual cross-layer SLA framework for cloud-hosted BDDA ii) a set of tailored QoS metrics that fit time sensitive BDDAs iii) a template of aSLAs that capture the complex inter-dependencies with pSLAs and iSLAs.

The main contribution of the *Q2* is an SLA-oriented modeling and simulation framework. It is well understood that simulation-based approaches to performance testing and benchmarking offer significant advantages. For example, multiple big data application developers and researchers can perform tests in a controllable and repeatable manner. Moreover, it is easier to find performance bottlenecks in a simulated environment than in real-world test beds. Further simulation-based approaches simplify experimentation with various hardware resource and big data processing framework configurations and collecting insights about the impact of each design choice on the QoS guarantees. In addition, developers and researchers can share their simulation datasets and environment setups, leading to better validation of hypothesis and reproducibility of results. Last but not least, instantiating multiple big data processing frameworks and diverse workload scenarios is feasible by using simulation and benchmarking approaches. Therefore, the proposed contribution will undoubtedly facilitate academics to study the SLA and QoS for cloud-hosted big data applications in a repeatable and controllable way.

VI. PRELIMINARY WORK

Since CloudSim is one of the widely used discrete-event simulation frameworks as it is highly extensible and flexible, I developed the BDDA platform simulator by extending the CloudSim. So far, I have completed implementation of a bare bone structure of Map/Reduce in Hadoop on top of CloudSim where we can imitate the behavior of map and reduce tasks. It follows a master-slave mode as does the real Hadoop. This is very important for our further research as it lays the foundation of modelling more complicated simulators for cloud-based big data applications. I have also enhanced CloudSim with new functionality so that it can support executing multiple CloudletLists sequentially. The work originated from a functionality limitation existing in current CLoudSim. Internally, the broker has a very simple operation and has a single cloudlet list, which means if you submitted multiple cloudlet lists to the broker, they are always merged to a single list, and they are handled as if only one submission were made. However, it is not suitable for a real MapReduce framework as the Reducer operation can only start after the corresponding Mapper operation.

REFERENCES

[1] L. Wang, R. Ranjan, J. Chen, and B. Benatallah, *Cloud computing: methodology, systems, and applications*: CRC Press, 2011.

[2] D. Abadi, R. Agrawal, A. Ailamaki, M. Balazinska, P. A. Bernstein, M. J. Carey, *et al.*, "The beckman report on database research," *ACM SIGMOD Record,* vol. 43, pp. 61-70, 2014.

[3] R. Zhang, R. Jain, P. Sarkar, and L. Rupprecht, "Getting Your Big Data Priorities Straight: A Demonstration of Priority-based QoS using Social-network-driven Stock Recommendation," *Proceedings of the VLDB endowment,* vol. 7, 2014.

[4] N. Marz and J. Warren, *Big Data: Principles and best practices of scalable realtime data systems*: O'Reilly Media, 2013.

[5] M. Kunjir, P. Kalmegh, and S. Babu, "Thoth: Towards managing a multi-system cluster," *Proceedings of the VLDB endowment,* vol. 7, 2014.

[6] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience,* vol. 41, pp. 23-50, 2011.

[7] M. B. Chhetri, Q. B. Vo, and R. Kowalczyk, "Policy-based automation of SLA establishment for cloud computing services," in *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*, 2012, pp. 164-171.

[8] S. Bouchenak, G. Chockler, H. Chockler, G. Gheorghe, N. Santos, and A. Shraer, "Verifying cloud services: present and future," *ACM SIGOPS Operating Systems Review,* vol. 47, pp. 6-19, 2013.

[9] S. K. Garg, S. Versteeg, and R. Buyya, "A framework for ranking of cloud computing services," *Future Generation Computer Systems,* vol. 29, pp. 1012-1023, 2013.

[10] M. Alrokayan, A. V. Dastjerdi, and R. Buyya, "SLA-aware Provisioning and Scheduling of Cloud Resources for Big Data Analytics."

[11] M.-C. Nita, C. Chilipirea, C. Dobre, and F. Pop, "A SLA-based method for big-data transfers with multi-criteria optimization constraints for IaaS," *2013 11th RoEduNet International Conference,* pp. 1-6, 2013.

[12] S. Govindan, J. Liu, A. Kansal, and A. Sivasubramaniam, "Cuanta: quantifying effects of shared on-chip resource interference for consolidated virtual machines," in *Proceedings of the 2nd ACM Symposium on Cloud Computing*, 2011, p. 22.

[13] R. Nathuji, A. Kansal, and A. Ghaffarkhah, "Q-clouds: managing performance interference effects for qos-aware clouds," in *Proceedings of the 5th European conference on Computer systems*, 2010, pp. 237-250.

[14] T. T. Q.Zhu, "a performance interface model for managing consolidated workloads in Qos-Aware Clouds."

[15] L. Wang, J. Zhan, C. Luo, Y. Zhu, Q. Yang, Y. He, *et al.*, "Bigdatabench: a big data benchmark suite from internet services," in *High Performance Computer Architecture (HPCA), 2014 IEEE 20th International Symposium on*, 2014, pp. 488-499.