Monitoring-as-a-Service In The Cloud

SPEC PhD Award (Invited Abstract)

Shicong Meng
Georgia Institute of Technology
smeng@cc.gatech.edu

Thesis Supervisor: Ling Liu
Georgia Institute of Technology
lingliu@cc.gatech.edu

ABSTRACT

State monitoring is a fundamental building block for Cloud services. The demand for providing state monitoring as services (MaaS) continues to grow and is evidenced by CloudWatch from Amazon EC2, which allows Cloud consumers to pay for monitoring a selection of performance metrics with coarse-grained periodical sampling of runtime states. This dissertation research [1], awarded with the SPEC Distinguished Dissertation Award 2012, is dedicated to innovative research and development of an elastic framework for providing state monitoring as a service (MaaS). We analyze limitations of existing techniques, systematically identify the need and the challenges at different layers of a Cloud monitoring service platform, and develop a suite of distributed monitoring techniques to support flexible monitoring infrastructure, cost-effective state monitoring and monitoring-enhanced Cloud management.

Categories and Subject Descriptors

D.4.7 [**Distributed Systems**]: Scalable Monitoring; C.2.4 [**Distributed Systems**]: Distributed applications

General Terms

Algorithms, Design, Performance, Reliability

Keywords

Cloud Monitoring, State Monitoring, Multi-Tenancy

1. INTRODUCTION

Today's Cloud datacenters are complex composition of large-scale servers, virtual machines, physical and virtual networks, middleware, applications, and services. Their growing scale and complexity challenge our ability to closely monitor the state of various entities, and to utilize voluminous monitoring data for better operation. Providing Monitoring-as-a-Service (MaaS)[5] brings a number of benefits to both Cloud providers and consumers.

First, MaaS minimizes the cost of ownership by leveraging the state of the art monitoring tools and functionalities. MaaS makes it easier for users to deploy state monitoring at different levels of Cloud services compared with developing ad-hoc monitoring tools or setting up dedicated monitoring hardware/software. Second, MaaS enables the pay-as-you-go utility model for state monitoring. This is especially important for users to enjoy full-featured monitoring services based on their monitoring needs and available budget. MaaS also brings Cloud service providers the opportunity to consolidate monitoring demands at different levels (infrastructure, platform, and application) to achieve efficient and scalable monitoring. Finally, MaaS pushes Cloud service providers to invest in state of the art monitoring technology and deliver continuous improvements on both monitoring service quality and performance. With the consolidated services and monitoring data, Cloud service providers can also develop value-add services for better Cloud environments and creating new revenue sources.

2. CONTRIBUTIONS

This dissertation research tackles critical challenges in MaaS with a layered approach that systematically addresses monitoring efficiency, scalability, reliability and utility at the monitoring infrastructure level, the monitoring functionality level and the monitoring data utility level. We analyze key limitations of existing techniques, and develop new techniques to offer more effective Cloud monitoring capabilities in this layered design. In addition, we built systems that help Cloud developers and users to access, process and utilize Cloud monitoring data. Specifically, this dissertation makes the following contributions.

2.1 Monitoring Infrastructure

At the monitoring infrastructure level, we propose REMO [3, 4] and Tide [6] which contribute to a Cloud-scale monitoring infrastructure that ensures the efficiency, scalability and multi-tenancy support of Cloud monitoring.

Monitoring Topology Planning [3, 4]. Large-scale monitoring can incur significant overhead on distributed nodes participating in collection and processing of monitoring data. Existing techniques that focus on monitoring task level efficiency often introduce heavily skewed workload distributions on monitoring nodes and cause excessive resource usage on certain nodes. We developed REMO, a resource-aware monitoring system that considers node-level resource constraints, e.g. monitoring-related CPU utilization should less than 5%, as the first-class factor for scheduling multiple monitoring tasks collectively. REMO optimizes the throughput of the entire monitoring network without causing excessive resource consumption on any participating node, which ensures performance isolation in multi-tenent monitoring environments. It also explores cost sharing opportunities among tasks to optimize monitoring efficiency. We deployed REMO on Sysem S, a large-scale distributed stream processing system built at IBM TJ Watson Lab. Through resource-aware planning, REMO achieves 35%-45% error reduction compared to existing techniques.

Self-Scaling Monitoring Infrastructure [6]. From traces collected in production datacenters, we found that monitoring and management workloads in Cloud datacenters tend to be highly volatile due to their on-demand usage model. Such workloads often make the management server a performance bottleneck. To address this problem, we developed Tide, a self-scaling management system which automatically scales up or down its capacity according to the observed workloads. We built the prototype of Tide by modifying VMware's vSphere management server and leveraging non-SQL Hadoop based HBase for scalable state persistence. The experimental results show that Tide provides consistent performance even with extreme volatile management workloads through self-scaling.

2.2 Monitoring Functionalities For Cloud

At the monitoring functionality level, we aim at providing new monitoring techniques to meet the unique and diverse Cloud monitoring needs, and we propose WISE [8, 7], Volley and CrystalBall [2] which deliver accurate, cost-effective and reliable monitoring results by employing novel distributed monitoring algorithms for error-prone Cloud environments.

Efficient Continuous State Violation Detection [7][8]. Most existing works on distributed state monitoring employ an instantaneous monitoring model, where the state is evaluated based on the most recent collected results, to simplify algorithm design. Such a model, however, tends to introduce false state alerts due to noises and outliers in monitoring data. To address this issue, we proposed WISE, window based state monitoring which utilizes temporal windows to capture continuous state violation in a distributed setting. WISE not only delivers the same results as those of a centralized monitoring system with a distributed implementation, but also decouples a global monitoring task into distributed local ones in a way that minimizes the overall communication cost.

Violation-Likelihood based Monitoring. Asynchronized monitoring techniques such as periodical sampling often introduce cost-accuracy dilemma, e.g., frequent polling state information may produce fine-grained monitoring data but may also introduce high sampling cost for tasks such as deep packet inspection based network monitoring. To address this issue, we proposed Volley, a violation likelihood based approach which dynamically tunes monitoring intensity based on the likelihood of detecting important results. More importantly, it always safeguards a user-specified accuracy goal while minimizing monitoring cost. Volley also coordinates sampling over distributed nodes to maintain the task-level accuracy, and leverages inter-task state correlation to optimize multi-task sampling scheduling. When deployed in a testbed datacenter environment with 800 virtual machines, Volley reduces monitoring overhead up to 90% with negligible accuracy loss.

Fault-Tolerant State Monitoring [2]. While we often assume monitoring results are trustworthy and monitoring services are reliable, such assumptions do not always hold, especially in large scale distributed environments such as datacenters where transient device/network failures are the norm rather than the exception. As a result, distributed state monitoring approaches that depend on reliable communication may produce inaccurate results with the presence of failures. We developed CrystalBall, a robust distributed state monitoring approach that produces reliable monitoring results by continuously updating the accuracy estimation of the current results based on observed failures. It also adapts to long-term failures by coordinating distributed monitoring tasks to minimize accuracy loss. Experimental results show that CrystalBall consistently improves monitoring accuracy even under severe message loss and delay.

2.3 Monitoring Enhanced Cloud Management

At the monitoring data utility level, we study intelligent techniques that utilize monitoring data to offer advanced monitoring management capabilities. As an initial attempt, we propose Prism which offers an innovative application provisioning functionality based on knowledge learned from cumulative monitoring data. We aim at utilizing multi-tier Cloud application performance data to guide application provisioning. Prism is a prediction-based provisioning framework that simplifies application provisioning by using performance prediction to find a proper provisioning plan for a performance goal in a huge space of candidate plans. As its unique feature, Prism isolates and captures the performance impact of different provisioning options, e.g., virtual machine types and cluster configurations, from performance monitoring data with off-the-shelf machine learning techniques. This technique avoids exploring the huge space of candidate provisioning plans with experiments. As a result, Prism can quickly find the most cost-effective plan with little cost for training performance prediction models.

Acknowledgement

This dissertation research was partially sponsored by grants from NSF NetSE and SaTC program, Intel ISTC on Cloud computing and an IBM PhD Fellowship.

3. REFERENCES

[1] S. Meng. Monitoring-as-a-service in the cloud. *PhD thesis, Georgia Institute of Technology*, 2012.

[2] S. Meng, A. Iyengar, I. Rouvellou, L. Liu, K. Lee, B. Palanisamy, and Y. Tang. Reliable state monitoring in cloud datacenters. In *IEEE CLOUD*, pages 951–958, 2012.

[3] S. Meng, S. R. Kashyap, C. Venkatramani, and L. Liu. Remo: Resource-aware application state monitoring for large-scale distributed systems. In *ICDCS*, pages 248–255, 2009.

[4] S. Meng, S. R. Kashyap, C. Venkatramani, and L. Liu. Resource-aware application state monitoring. *IEEE Trans. Parallel Distrib. Syst. (TPDS)*, 23(12):2315–2329, 2012.

[5] S. Meng and L. Liu. Enhanced monitoring-as-a-service for effective cloud management. *IEEE Transactions on Computers (TC)*, to appear.

[6] S. Meng, L. Liu, and V. Soundararajan. Tide: Achieving self-scaling in virtualized datacenter management middleware. In *Middleware'10*.

[7] S. Meng, L. Liu, and T. Wang. State monitoring in cloud datacenters. *IEEE Trans. Knowl. Data Eng. (TKDE)*, 23(9):1328–1344, 2011.

[8] S. Meng, T. Wang, and L. Liu. Monitoring continuous state violation in datacenters: Exploring the time dimension. In *ICDE*, pages 968–979, 2010.