Understanding Performance Interference Benchmarking and Application Profiling Techniques for Cloud-hosted Latency-Sensitive Applications

Shashank Shekhar
Dept of EECS, Vanderbilt University
Nashville, Tennessee, USA
shashank.shekhar@vanderbilt.edu

Yogesh Barve
Dept of EECS, Vanderbilt University
Nashville, Tennessee, USA
yogesh.d.barve@vanderbilt.edu

Aniruddha Gokhale
Dept of EECS, Vanderbilt University
Nashville, Tennessee, USA
a.gokhale@vanderbilt.edu

ABSTRACT

Modern data centers are composed of heterogeneous servers with different architectures, processor counts, number of cores and speed. They also exhibit variability in memory speed and size, storage type and size and network connectivity. In addition, the servers are multi-tenant, often hosting latency sensitive applications in addition to the traditional batch processing applications. To provide bounded and predictable latencies, it is necessary for the cloud providers to understand the performance interplay among the co-hosted applications. To that end, we present our integrated and extensible framework called INDICES for users to conduct a variety of performance benchmarking experiments on multi-tenant servers. The framework also performs centralized data collection for a range of resource usage and application performance statistics in order to model the performance interference and estimate the execution times for the cloud hosted applications.

CCS CONCEPTS

• **Computer systems organization** → **Cloud computing**; • **Computing methodologies** → *Distributed algorithms*;

KEYWORDS

Cloud Computing; Performancing Monitoring; Cloud Benchmark; Performance Interference; Resource Management

1 INTRODUCTION

Cloud infrastructure providers must have an up to date understanding of the usage of their cloud resources so that they can effectively manage their cloud platforms while supporting multi-tenancy. At the same time, timely and scalable access to various resource usage statistics is critical to service providers, who host their services in the cloud. This is required in order to ensure that their applications provide the required quality of service to their customers through elastic and on demand auto-scaling while minimizing service-hosting costs. Thus, these providers must understand how their services will perform under a variety of multi-tenancy scenarios and workload patterns. Conducting such benchmarking experiments and obtaining the desired resource statistics to pinpoint the sources of problems, such as the level of performance interference, is a hard problem.

Statistics collection in the cloud is a hard problem for a variety of reasons including multi-tenancy, heterogeneity in hardware and operating systems, and availability of hardware-specific, low-level statistics collection tools all of which make it extremely complex for providers to use existing capabilities and extend them as hardware changes and the statistics collection needs change. These challenges are further amplified as cloud platforms increasingly span fog and edge resources. Thus, a framework that is extensible and provides a higher level of abstraction to make it easy to use is needed.

Although, data collection tools such as collectd [5] and benchmarking frameworks such as CloudSuite [4], PARSEC [1] and YCSB [2] are designed to benchmark cloud applications and collect metrics, they do not focus on modeling performance interference on multi-tenant heterogeneous servers. On the other hand, a benchmark like iBench [3] tries to quantify the data-center performance interference, but they provide only some of the building blocks, thereby making the users responsible to develop and integrate the capabilities and deal with the complexities of low level details.

To overcome these challenges, this tutorial presents a framework called INDICES (INtelligent Deployment for ubIquitous Cloud and Edge Services) [6], which builds on collectd and provides an integrated and extensible framework for users to conduct a variety of performance benchmarking experiments and collect a range of resource usage and application performance statistics. Thus, INDICES enables the performance modeling of cloud-hosted applications on heterogeneous multi-tenant servers.

The rest of the paper is organized as follows: Section 2 presents the framework description; Section 3 describes the tutorial organization; and finally Section 4 presents concluding remarks.

2 FRAMEWORK DESCRIPTION

The INDICES framework consists of multiple components. The primary component is the data collection framework, which is built using the collectd [5] monitoring tool which has a plugin-based extensible architecture. We developed plugins to collect micro-architectural metrics, which are necessary for accurate performance interference modeling. In addition, the framework also collects virtual machine (VM) or Docker container-specific metrics. The stand-alone metric collection framework was integrated with InfluxDB time-series database such that centralized data collection can be

UCC '17: 10th International Conference on Utility and Cloud Computing, December 5–8, 2017, Austin, TX, USA

ACM ISBN 978-1-4503-5149-2/17/12.
https://doi.org/10.1145/3147213.3149453

performed and aggregated. Figure 1 depicts the data collection architecture.

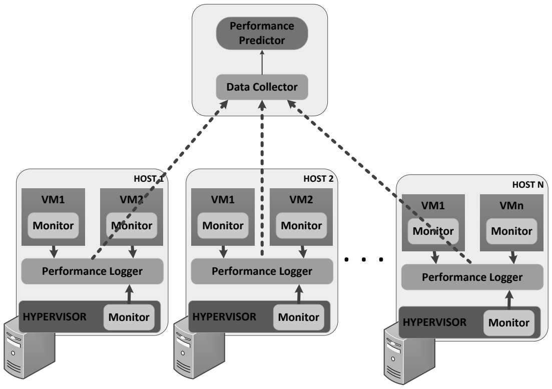

Figure 1: INDICES Data Collection Architecture

The other key component of the framework is the benchmarking component. This consists of a number of latency-sensitive client-server target applications. In addition, there are other workload applications that are used to cause performance interference on the target applications. The final component is the performance modeling component. Using the collected metrics, it helps in developing the application sensitivity and pressure as described in our prior work [6].

3 TUTORIAL ORGANIZATION

The tutorial has been divided into two sections. First we introduce the INDICES framework and explain the design details. This session includes the following:

- Motivation of the subject by highlighting the importance of resource monitoring in distributed and cloud systems alluding to the complexities in monitoring of system performance.
- Highlighting the causes of application performance interference and its relevance in the cloud, fog and edge computing realm.
- Introduction to the design of the INDICES framework and its underlying technologies.
- Short demo of INDICES used in application performance benchmarking.

In the second half, the audience will perform hands-on activities that include:

- INDICES Benchmarking-I: Configuration & Setup
- INDICES Benchmarking-II. Execution and deployment of the framework for benchmarking and monitoring of a target application under various collocated benchmarking applications (e.g: PARSEC, STREAM)
- Discussions about the results.

Using INDICES framework, one can quickly observe performance characteristics of applications deployed under various computing environments with heterogeneous resources. Moreover, using the INDICES framework one can test the target applications' performance interference avoidance strategies which are inherent factors resulting in poor application performance in multi-tenant distributed systems. By using a concrete example from the PARSEC-based benchmarking tool to observe performance interference effects in distributed systems, the audience will discover the ease of use, detailed system metrics logging and visualization benefits from using the INDICES framework in a practical and research problem.

Upon completing this tutorial, attendees will be able to:

- Recognize the inherent and accidental complexities involved in designing and developing applications for distributed systems.
- Gain knowledge as to how the INDICES framework can help alleviate application interference issues by better designing applications to be tolerant to such system behavior.
- Acquire hands-on knowledge on how to use the INDICES framework for benchmarking and monitoring application performance interference and other system characteristics.

4 CONCLUSIONS

This paper described an extensible benchmarking, and metrics collection and analysis framework called INDICES, which is useful in determining performance interference in cloud data centers and fog/edge environments alike. In turn these metrics can serve to make effective dynamic resource management decisions in the cloud. The source code for the INDICES framework is available at: https://github.com/doc-vu/indices.

ACKNOWLEDGMENTS

This work was funded by NSF CNS US Ignite 1531079. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF.

REFERENCES

[1] Christian Bienia, Sanjeev Kumar, Jaswinder Pal Singh, and Kai Li. 2008. The PARSEC benchmark suite: characterization and architectural implications. In *17th international conference on Parallel architectures and compilation techniques*. ACM, 72–81.

[2] Brian F Cooper, Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, and Russell Sears. 2010. Benchmarking cloud serving systems with YCSB. In *Proceedings of the 1st ACM symposium on Cloud computing*. ACM, 143–154.

[3] Christina Delimitrou and Christos Kozyrakis. 2013. ibench: Quantifying interference for datacenter applications. In *Workload Characterization (IISWC), 2013 IEEE International Symposium on*. IEEE, 23–33.

[4] Michael Ferdman, Almutaz Adileh, Yusuf Onur Koçberber, Stavros Volos, Mohammad Alisafaee, Djordje Jevdjic, Cansu Kaynak, Adrian Daniel Popescu, Anastasia Ailamaki, and Babak Falsafi. 2012. Clearing the clouds: a study of emerging scale-out workloads on modern hardware. In *Proceedings of the 17th International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2012, London, UK, March 3-7, 2012*. 37–48. https://doi.org/10.1145/2150976.2150982

[5] Florian Forster. 2017. Collectd - The System Statistics Collection Daemon. http://collectd.org. (2017).

[6] Shashank Shekhar, Ajay Chhokra, Anirban Bhattacharjee, Guillaume Aupy, and Aniruddha Gokhale. 2017. INDICES: Exploiting Edge Resources for Performance-Aware Cloud-Hosted Services. In *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*. Madrid, Spain, 75–80. https://doi.org/10.1109/ICFEC.2017.16