QoS and Performance Metrics for Container-based Virtualization in Cloud Environments

Paolo Bellavista *et. al.*

PhD, Professor

Mobile Middleware Research Group

Dipartimento di Informatica – Scienza e Ingegneria, **University of Bologna**, Viale Risorgimento 2, 40136 Bologna, Italy

paolo.bellavista@unibo.it

20th International Conference on Distributed Computing and Networking,
Indian Institute of Science, Banglore , India, 4-7, January 2019

ICDCN 2019

Agenda

Introduction → Performance Metrics → Results → Conclusions

Introduction

Motivation

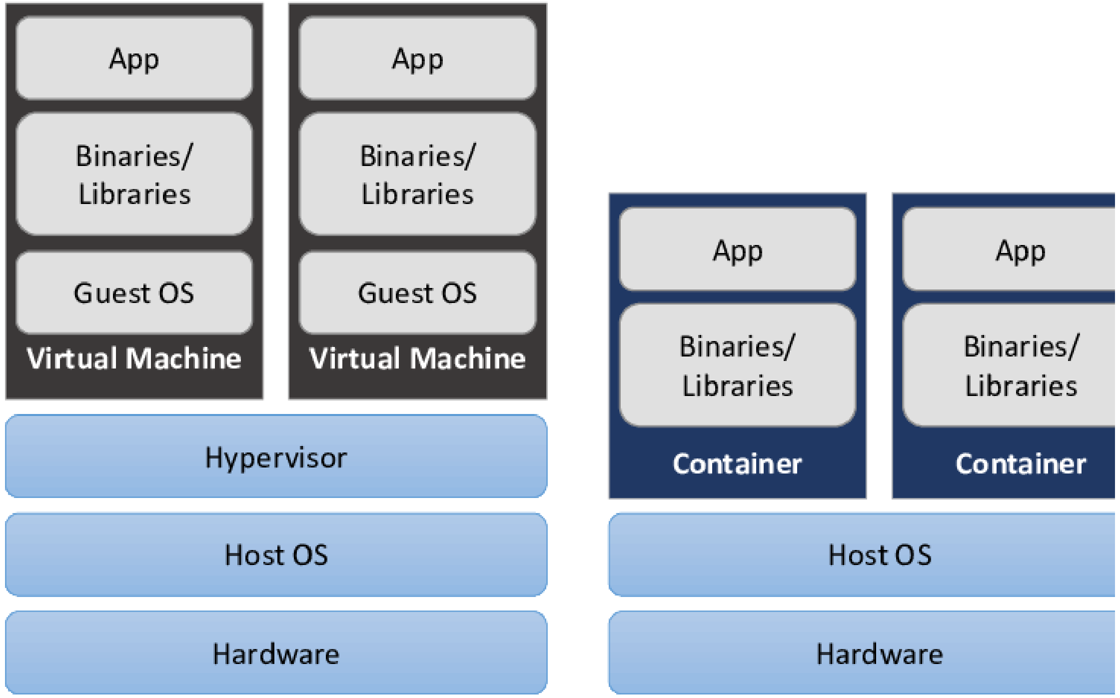
- Cloud architectures emerged to overcome performance shortcomings of server-based counterparts
- In Cloud, bare-metals share their capacities, thus providing a transparent layer (hiding the logistics of handling the underlying physical infrastructure) for big computing workloads
- Managing those commodities in a cohesive and robust manner requires costly consolidation
- Virtualization technologies emerged as a solution

CLOUD BASED VERSUS **SERVER BASED**

Cloud Based	Server Based
Cloud refers to a shared pool of computing resources that provides on-demand access to these resources via the Internet.	Server refers to a dedicated computer which manages access to centralized resources in a network.
Cloud is based on Infrastructure-as-a-Service (IaaS) model that provides virtualized computing resources over the internet.	Server based computing refers to the technology where applications are implemented and controlled on the server.
A cloud based application is any software program or application that operates in the cloud space.	A server based application refers to a program or application stored on a remote server.

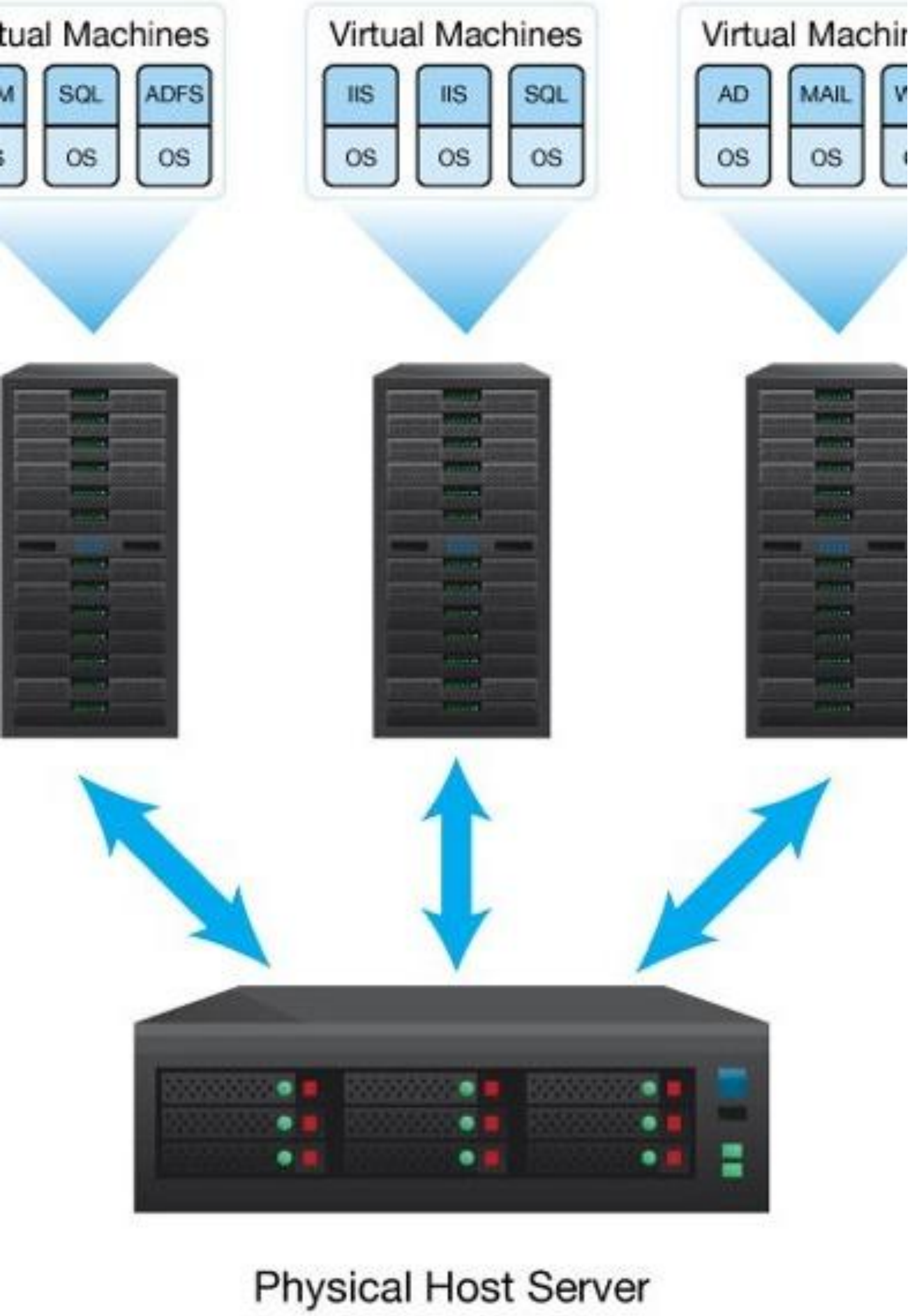
Virtualization

- Virtualization technologies abstract the underlying complexities of network system logistics and resource management mechanisms
- Efficient resource utilization by concurrently run multiple operating systems on same server
- Two primary kinds of virtualization
  - **Hypervisor-based virtualization**. Induce performance overhead.
  - **Containerization**. A lightweight virtualization.
- Which one to opt for? Metrics are needed to decide.



Hypervisor-based

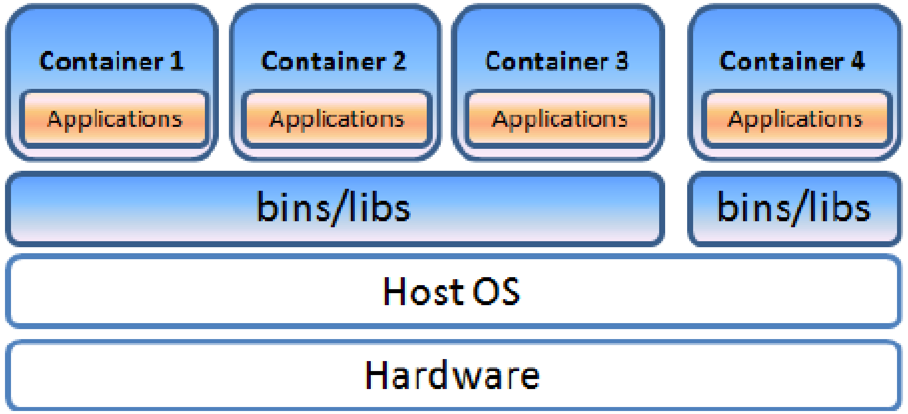
Container-based



Server virtualization

- De facto standard in cloud computing
- Allows running multiple OS instances concurrently on top of a single (or more) bare-metal server(s), dynamically partitioning and sharing available physical resources (CPU, storage, memory and IO)
- We focus on Virtual Machines, a hypervisor-based virtualization of two types:
  - **Bare-metal-based**, where a hypervisor sits directly on top of underlying hardware and
  - **Hosted version**, where a hypervisor stands on top of a host operating system.
- A clear shortcoming , requires loading full operating system to each VM, which quickly become laborious

Containerization



- Abstracts workloads from the underlying hardware, allowing multiple isolated systems (a.k.a containers) run on top of a single kernel
- Containers encapsulate only components that are necessary to run a specific software
- Alternatives include Linux VServer, OpenVZ, and LXC
- Linux containers (LXC) provides a lightweight operating system virtualization
- We focus on LXD (patterned after LXC), which aims at delivering a multi-host container management with advanced features.

QoS-aware & Performance Metrics for Cloud Containerization

CPU Analysis

- Two benchmarks
  - **CPU-power**. Execution time of a single process that exploits the number of existing cores through the concept of multi-threading.
  - **CPU-contention**. Analyzing the behavior of execution time by operating together multiple compute instances that compete to access the same shared computing resources.

Network Analysis

- Measures network service quality, including **bandwidth**, **latency**, and **throughput**
- For the throughput and bandwidth, we use **iperf**
- Tests include
  - **InterCloud**, a computing machine runs inside a cluster, whereas an Iperf server executes outside
  - **IntraCloud**, a computing machine runs inside the cluster and an Iperf Server executes in the same cluster but on a different physical server host
  - **IntraNode**, a computing machine runs inside a cluster and an Iperf Server is another computing machine that runs on the same physical server host

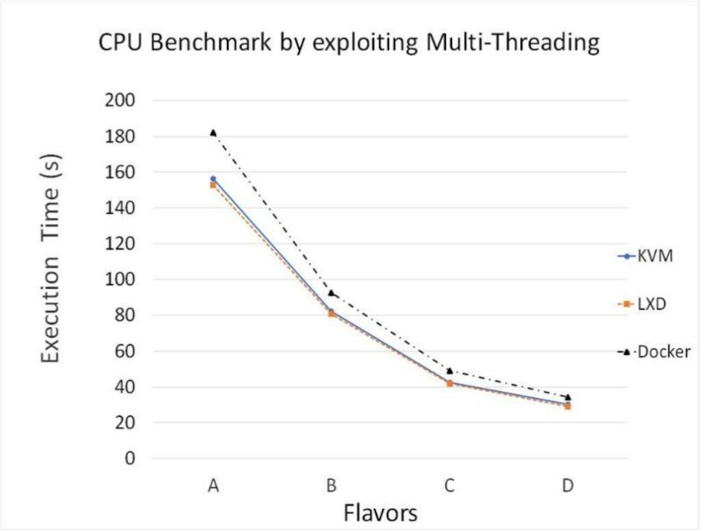
Read/write Analysis

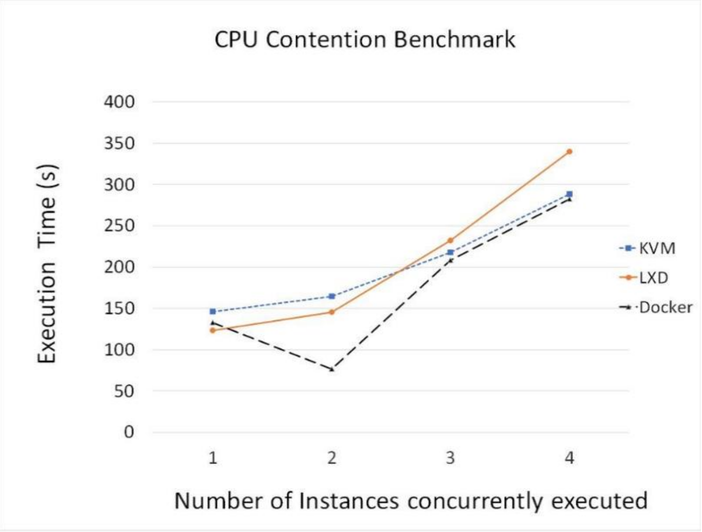
- Evaluates IO access patterns within each compute instance
- We employ Bonnie++ benchmark for stress testing using application data sizes that are significantly larger than the system memory
- Computing machines are set to 512MB of RAM and the files that they are working on have a size of 24GB (forty-eight times the size of the RAM)

Complimentary Analysis

- **Density Analysis.** Measures the number of compute instances that a server can host without degrading its performance
- **Bootstrap Analysis.** Measures the time needed to provide a complete service instance, booting it up and completing a snapshot instantiation

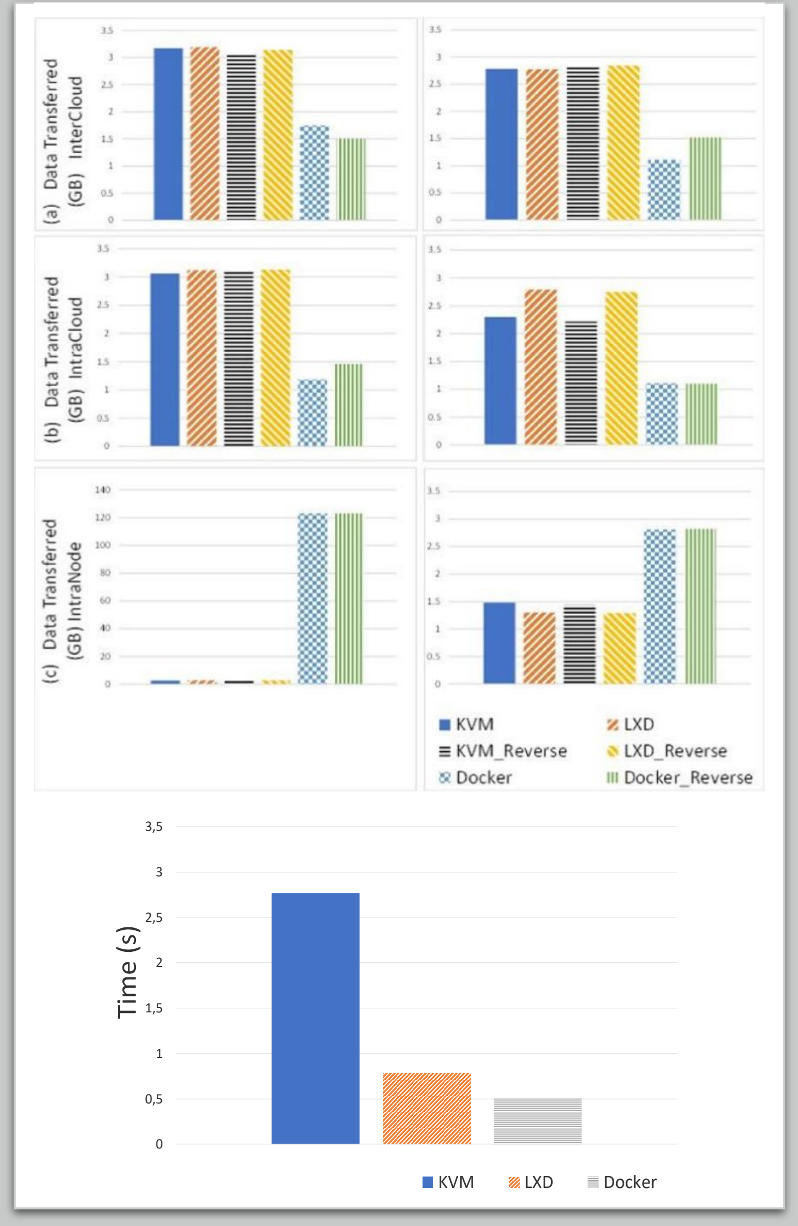
Results & Discussion





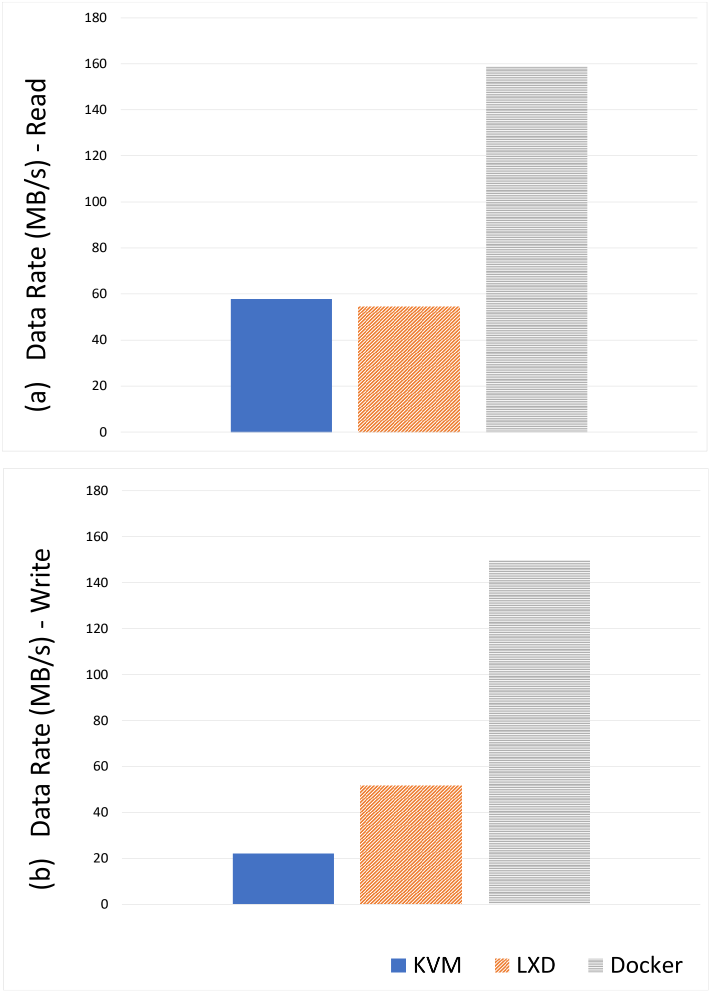
CPU Analysis

- We employ four flavor sets, all containing 2048 MB RAM and 20 GB disk, while alternating the number of vCPUs between 1,2,4, and 6 for A, B, C, and D flavors respectively
- For Docker, we employ Rancher to limit the resource capabilities of each Docker container
- LXD outperform KVM by a tiny fraction. However, Docker performs unfavorably in this case



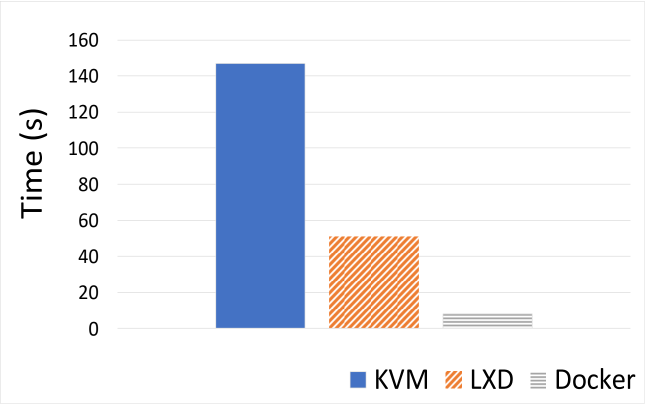
Network Analysis

- Docker outperforms counterparts for **intra-node** scenarios in the language of TCP and UDP throughput
- However, KVM has shown to have better performance in case of **intracloud**, and LXD in case of **intercloud**
- server-based virtualization imposes a significant overhead in terms of latency (in case of **intercloud**, similar trends also occurred in **intracloud** and **intra-node**) as LXD containerization and Docker outperforms KVM by orders of magnitude
- **Despite** that containerization outperforms counterparts for same cluster deployments , they behave less effectively when crossing boundaries of their cluster



Input/output

- Docker achieved the best performance with IO benchmarks. It outperforms KVM hypervisor server virtualization in addition to LXD containerization by at least 2x, which is quite significant
- The OpenStack KVM-based instance outperforms LXD by a tiny fraction for write operations and almost equivalent to that of LXD for read



Bootstrap

- Container-based virtualization outperforms **hypervisor** counterparts in term of provisioning time
- Attributed to the fact that **VM** require loading the full OS, counteracting the benefits of virtualization

Conclusions

Concluding remarks

- We have designed metrics for measuring QoS and performance of containerization in the cloud, aiming at guiding the selection of the right virtualization solution for cloud deployments
- Despite that they excel at some points, containers are not a fit-for-all alternative that may ultimately replace traditional contemporaries
- Performance of virtualization technologies is dependent upon the mixture of configurations used for stress-testing, including memory size, CPU power, and network configuration

QoS and Performance Metrics for Container-based Virtualization in Cloud Environments

Thank you !

Paolo Bellavista, Ph.D., Professor
Email: paolo.bellavista@unibo.it
Web: http://lia.disi.unibo.it/Staff/PaoloBellavista/

Questions and Discussion