Comparison of Virtualization and Containerization Techniques for High Performance Computing

Yuyu Zhou*†, Balaji Subramaniam†, Kate Keahey†, John Lange*
*Department of Computer Science, University of Pittsburgh
†Mathematics and Computer Science Division, Argonne National Lab

NIMBUS

Motivation and Background

- Traditional High Performance Computing (HPC)
  - Mainly use dedicated, in-house clusters
- Cloud computing is attractive for HPC users
  - Cloud provides several advantages
    - Example: lower queue waiting time
  - Large cloud companies cater to HPC users
    - Amazon provides HPC instances

HPC User Concerns

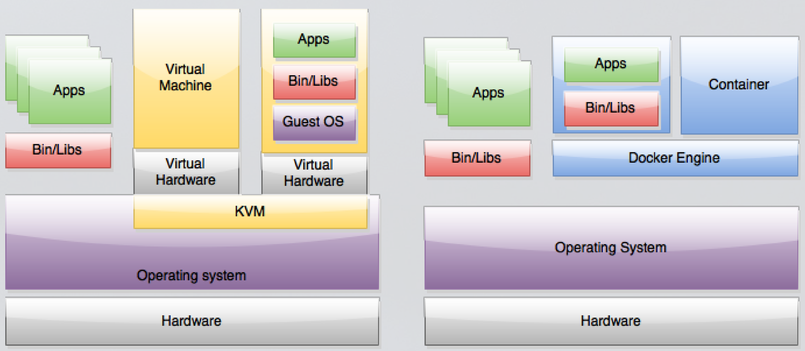
- Functional differences between cloud and dedicated clusters
  - What does the user see when using a cloud instance vs. bare-metal instance?
- Performance overhead associated with cloud
  - VM or Container usage introduces overhead
- Virtual Instances co-location leads to prohibitive performance viability

Research Agenda

- Study virtualization and containerization features
- Performance analysis
  - Micro-benchmarks
  - Mini-applications
    - Scale-up
    - Scale-out

Platforms Used

- KVM to study virtualization
- Docker to study containerization



High level description of KVM (left) and Docker (right)

Feature Comparison

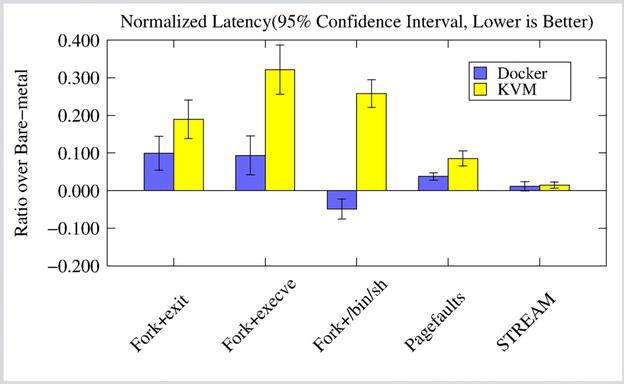
	KVM	Docker
Guest OS	Windows / Linux / Unix	Linux with same Kernel
Startup Time	VMs take a few minutes to boot up	Containers take a few seconds to boot up
Isolation and Security	VMs are fully isolated. The attack surface is VMM	The attack surface for the containers is the "shared OS kernel"
Live Migration Support	Yes	No (pre-alpha level support available)
Integrated with OpenStack	Yes	Yes

Experiment Environment

- Chameleon: a bare-metal reconfigurable environment
- Compute node configuration
  - 24 cores with two sockets (without hyper-threading)
  - 128 GB of memory

Micro-benchmark Results

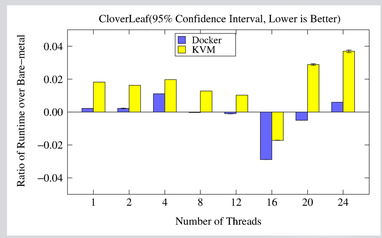
- lmbench: micro-benchmarks to test performance of simple operations (e.g. process creation, system call, memory read)

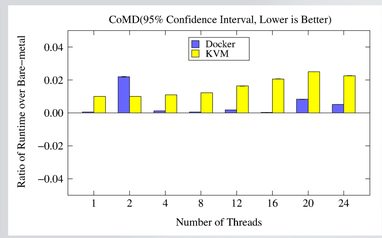


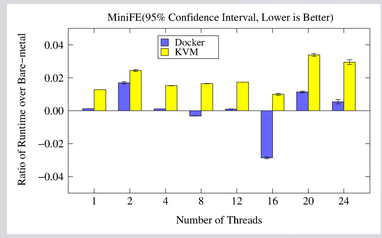
- For most of the benchmarks, KVM has worse performance than Docker

Scale-Up Experiment Results

- Scale-up experiments on single node
- OpenMP version of three mini-applications
  - Hydrodynamics, molecular dynamics and partial differential equations.



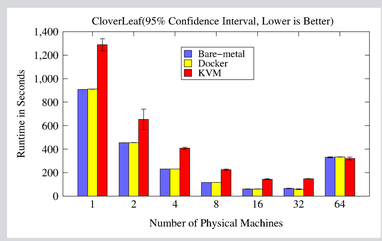


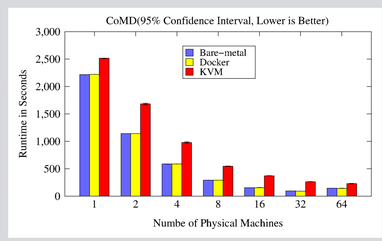


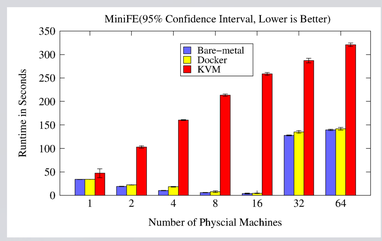
- KVM has 1.57%, 1.60% and 2.00% average performance overhead , respectively, for CloverLeaf, CoMD and MiniFE.
- Docker has -0.17%, 0.50% and 0.07% average performance overhead, respectively, for CloverLeaf, CoMD and MiniFE.

Scale-Out Experiment Results

- Scale-out experiments on sixty-four nodes
- MPI version of three mini-applications







- KVM has 73.77%, 84.85% and 1789.74% average performance overhead, respectively, for CloverLeaf, CoMD and MiniFE.
- Docker has -1.49%, -0.41% and 17.90% average performance overhead, respectively, for CloverLeaf, CoMD and MiniFE.

Future Work

- Scale out to more nodes
- Investigate the root reason for the results
  - Measure low-level performance counters
- Investigate how co-location affects performance

Acknowledgements

Argonne NATIONAL LABORATORY
University of Pittsburgh
Chameleon