

Docker Performance Benchmarking

Bare Metal vs Hypervisor

Docker Day
Duke University
September 11, 2014

Why?

Because I'm Lazy



Why?

Hypervisor convenience vs Bare Metal performance



vs



What?

Benchmark!

Based on:

*KVM and Docker LXC Benchmarking on OpenStack**

- Serial Container Boot
- Steady-state Container Packing
- CPU Performance
- MySQL Transactions
- MySQL Indexing and Queries
- File I/O Operations
- Memory Performance
- Application Performance - Blogbench (Simulated Blog)
- Application Performance - Apache Bench (Raw Served Webpages)

What?

Hypothesis:

“Bare Metal will beat the pants off of Hypervisor”



How?

Physical Server

- Cisco UCS Blade CCSB-B200-M3
- Intel E5-2665@2.4 GHz (8 Core)
- 256 GB 1600 RAM
- 2 Toshiba 300GB SAS 15K RPM

Host OS

- CoreOS
- Beta 324.5.0

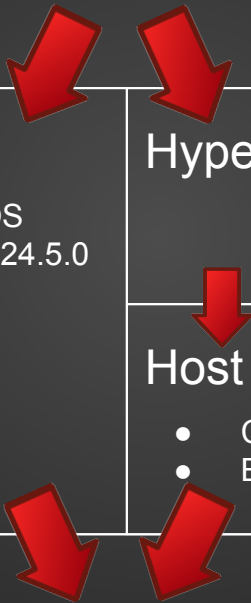
Hypervisor

Host OS

- CoreOS
- Beta 324.5.0

Docker

- 0.11.1, Build fb99f99



How?

Process:

- Create Docker images for tests
- Push images to Duke Docker repo
- PXE Boot CoreOS
- Pull runner scripts via CoreOS cloud-config
- Run runner scripts as root
- Collect data via ssh tunnel to external server

Results!

Hypothesis:

*“Bare Metal will beat the
pants off Hypervisor”*



RESULTS!

Only 7 out of 12 pants conclusively beaten off



Bare Metal Wins:

- MySQL Transactions
- MySQL Indexing and Queries
- Memory Performance (all three)
- Blogbench (Writes only)
- Apache Bench

RESULTS!

Only 3 out of 12 pants tied for performance*

Hypervisor Ties Bare Metal:

- Serial Boot Tests
- Steady State Packing
- File I/O

*Google doesn't have any funny images for this phrase. Who would have thought?

RESULTS!

Surprise! Hypervisor Beats Bare Metal in 2 (or 4)



Hypervisor Wins:

- CPU Performance
- Blogbench (Reads only)

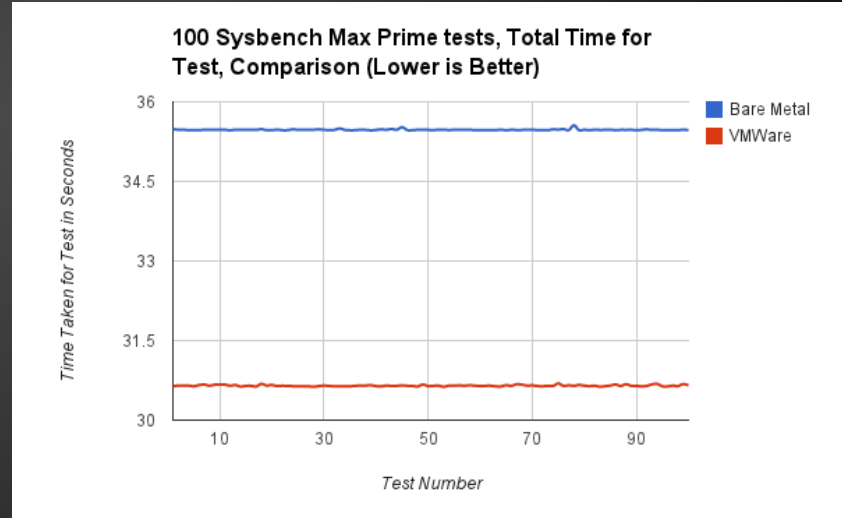
Hypervisor (sort of) Wins:

- Serial Boot Tests (I/O Wait)
- Steady State Packing (I/O Wait)

(The Interesting) Details

CPU Max Prime Number Calculations

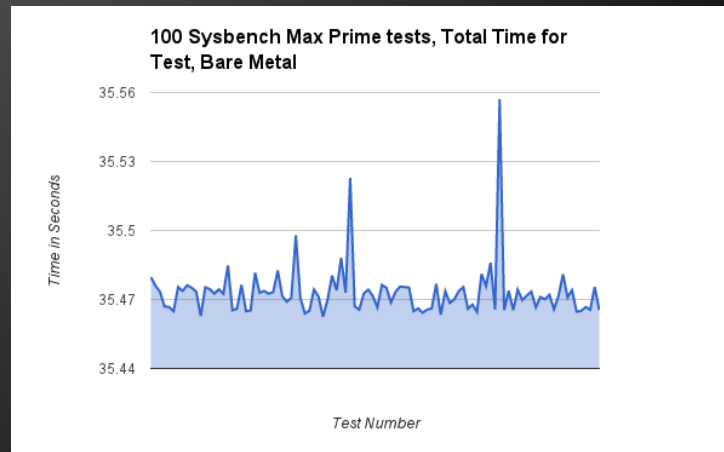
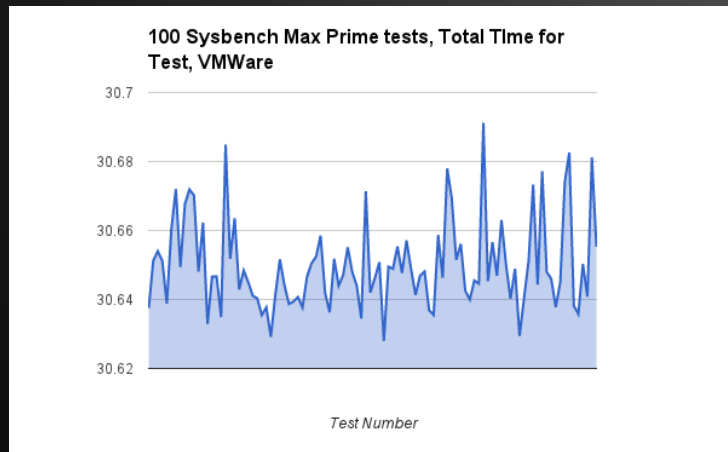
Hypervisor is faster at
Prime Number
calculations.



(The Interesting) Details

CPU Max Prime Number Calculations

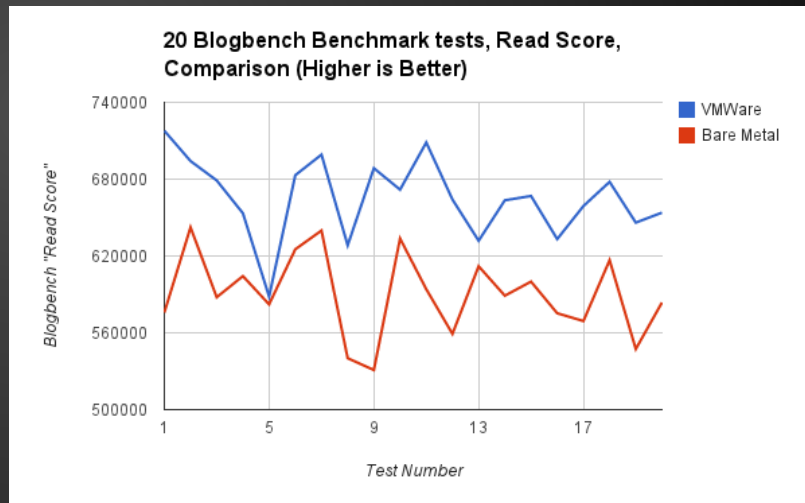
Hypervisor is (slightly) more consistent at Prime Number calculations
(note the scale)



(The Interesting) Details

Blog Bench (simulated blog) Reads

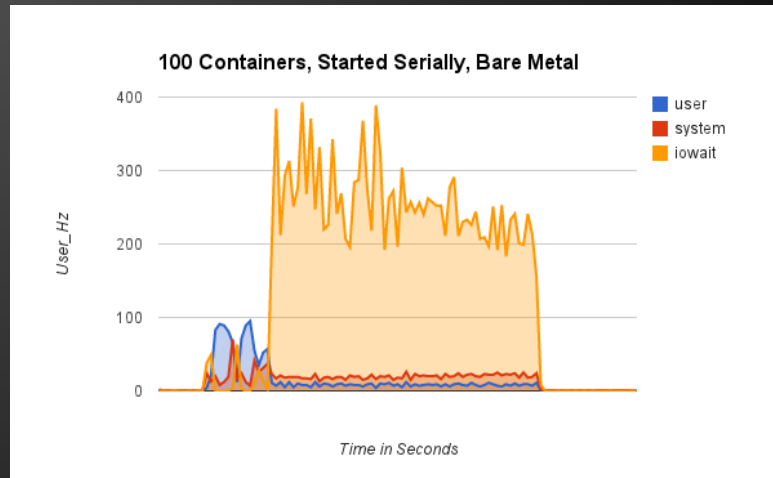
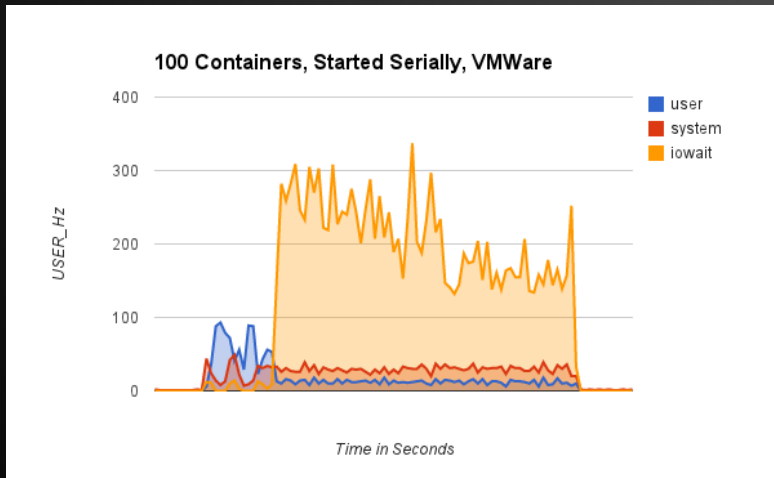
Hypervisor is faster at
(simulated) blog
reads.



(The Interesting) Details

Serial Container Boot

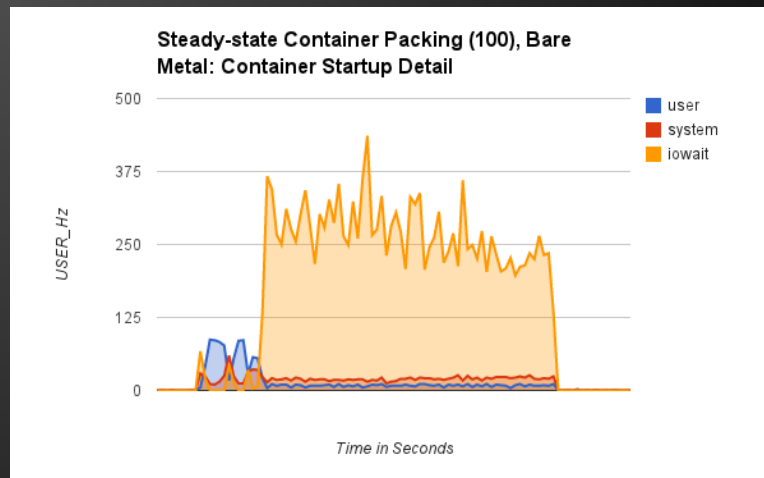
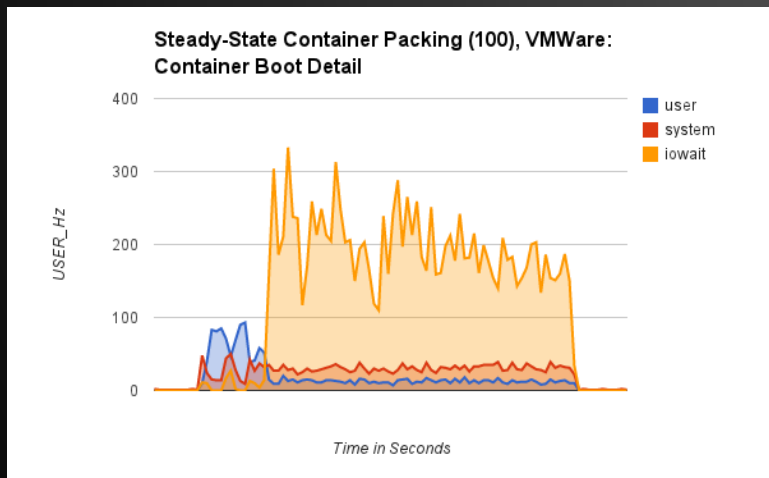
Negligible difference in CPU, but Hypervisor has (slightly) better I/O Wait



(The Interesting) Details

Steady-State Packing

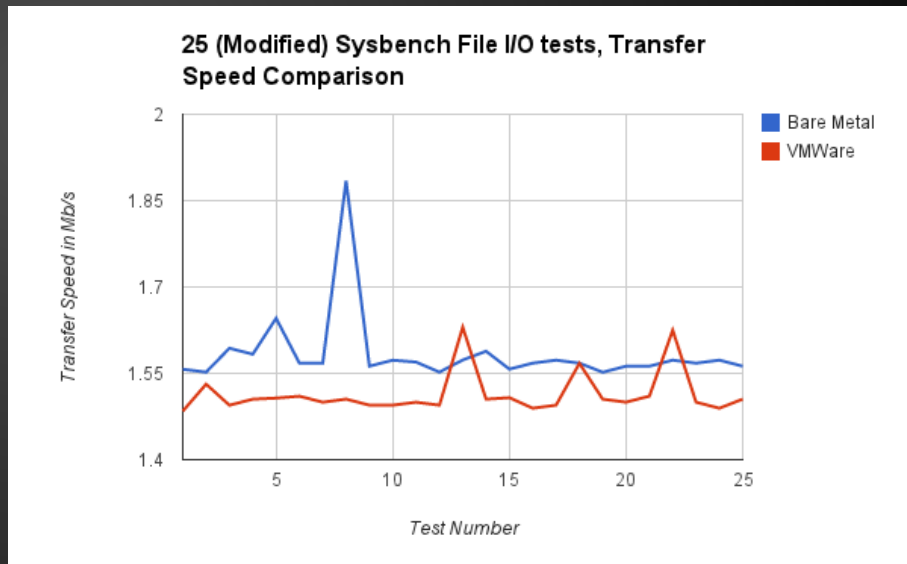
Same results as Serial Container Boot;
basically the same test with more normalization



(The Interesting) Details

File I/O Test

Despite previous test results, the difference in File I/O here is negligible.



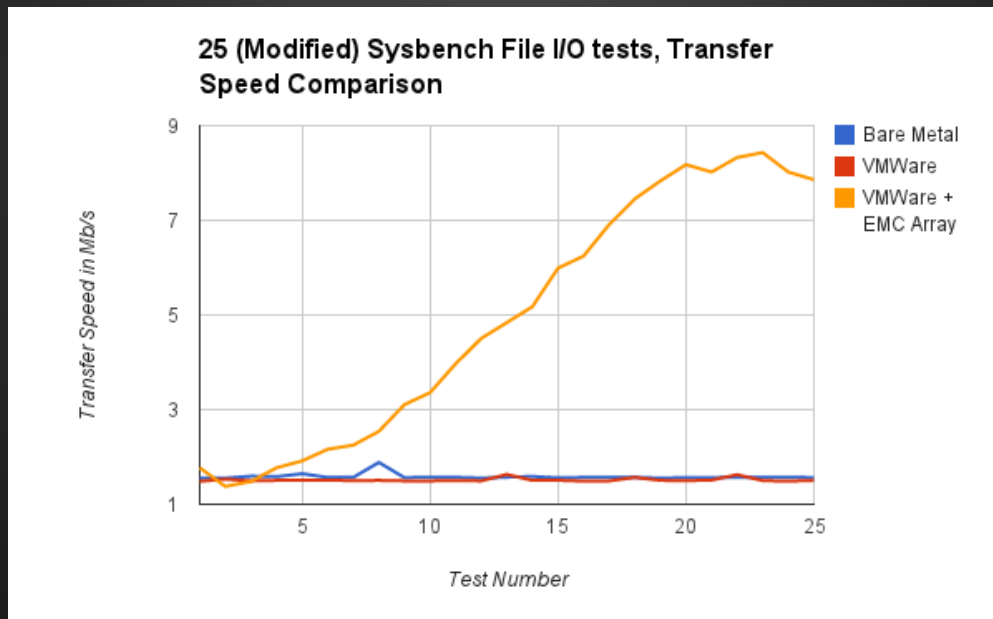
Tangential Learning Points

Unrelated to testing the hypothesis, I learned:

- The addition of Hypervisor has almost no effect on Docker-related tasks
 - Container Boot
 - Container Shutdown
 - Idle Containers
- Disk I/O is the largest bottleneck in Docker-related tasks
 - Considerably more than CPU or RAM
- Hypervisor gets a much larger boost in RAM performance by having more physical RAM to write to, even if the same amount is presented to the VM
- Enterprise storage arrays that detect “Hot” or active write blocks and adjust accordingly can have a HUGE impact on File I/O

Tangential Learning Points

Re: Enterprise Storage “hot block” detection



Final Thoughts

I do not trust these results

- Sample Size Too Small
- Surely missed something
- Surely misinterpreted something



Final Thoughts

I do not trust these results

Please help:

- Review my method, results
- Replicate in your own environment
- Let me know what you find out



For Science!

Info

GitHub Repo:

- Method
- Results
- Docker Images
- CoreOS Cloud-config.yml
- Benchmark running scripts

<https://github.com/DockerDemos/vm-docker-bench>

Info

Me:

- Chris Collins
- collins.christopher@gmail.com
- <https://github.com/clcollins>