Use of containerisation as an alternative to full virtualisation in grid environments.

Robin Long and Roger Jones
Lancaster University

1. Introduction

Virtualisation allows for a more dynamic and efficient use of the grid. By using virtualisation, multiple independent servers may be provisoned on a single node, or even multiple operating systems. Such a setup allows for a rapid reprovisioning of services such as conversion from a High Level Trigger farm to a traditional grid site.

Traditional methods require the virtualisation of the hardware and kernel which causes significant overhead. By negating the need to virtualise the kernel and hardward, significant savings can be made in overheads.

2. Containers

Traditional virtualisation and paravirtualisation requires the whole operating system and hardware to be virtualised. Using containers, individual applications can be separated from each other and "virtual machines" can be created but without the overhead of traditional setups.

Containers allow the running of individual applications and their dependencies. These, in the case of docker, then run on top of the docker services. Only the userland and applications are virtualised which allows significant overhead savings. This is detailed in Figure 1.

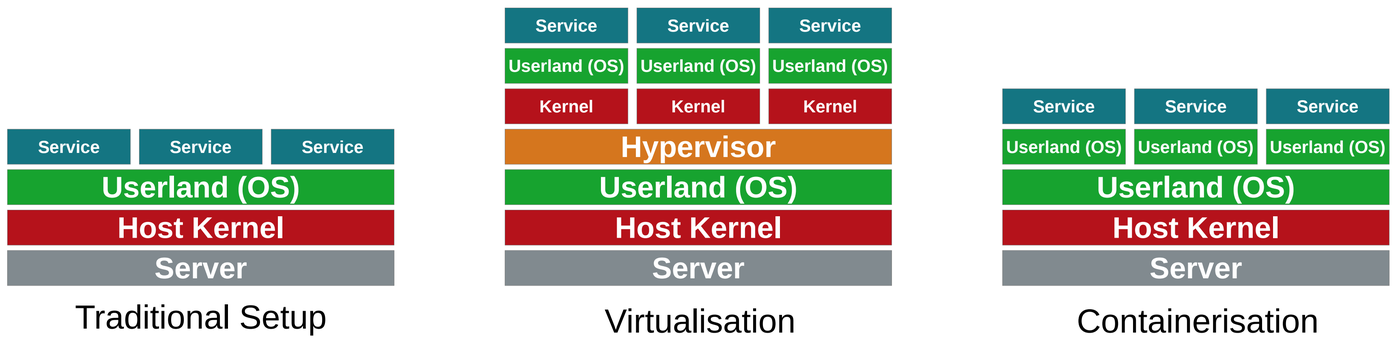


Figure 1: A graphical representation of how baremetal, virtualisation (kvm), and containerisation (docker) works. To run a service on a barmetal machine, we need the kernel, userland (libraries that interact with the kernel), and the software we want to run. When we create a virtual machine, we need to virtualise all of this. Using containers, we only need to virtualise the userland and software as the containers' userland interacts with the host kernel.

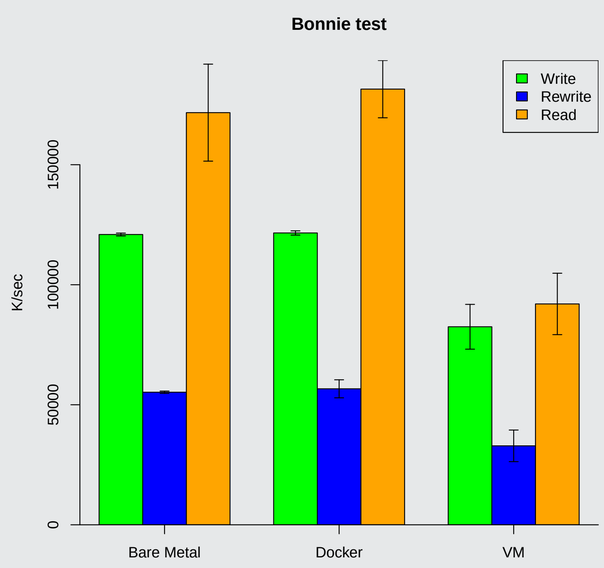
3. The Analysis

In this analysis we look at how virtualisation and containerisation affect the performance of a machine. Comparisons are made between a standard physical machine running one operating system (64GB RAM, 16 Cores), a virtual machine using all the resources, and a container doing the same. We have used Scientific Linux 6 for the OS, kvm for the virtual machine, and docker for the container.

The performance of each of these is then tested using standard benchmarking tools; Bonnie++, HEPSPEC, and Iperf. All tests were run three times.

4. Bonnie++

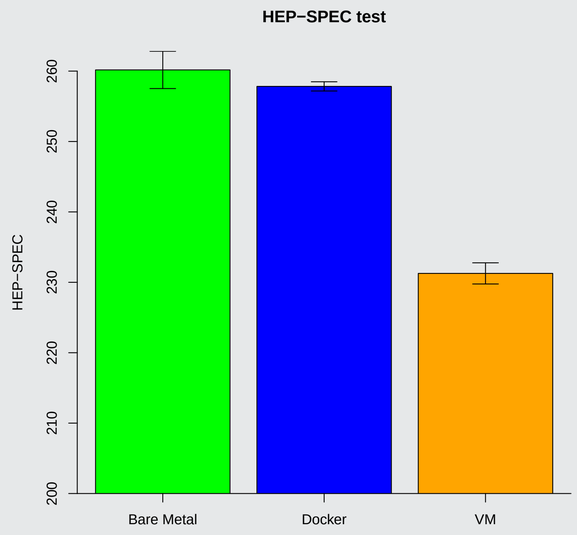
Bonnie++ is designed to benchmark the hard drive and file system performance.



Bonnie provides the most interesting results. A significant drop in all areas (write, re-write, and read) is seen for the full virtualisation, whilst no significant drop is seen in any performance for containerisation when compared to bare metal.

5. HEPSPEC

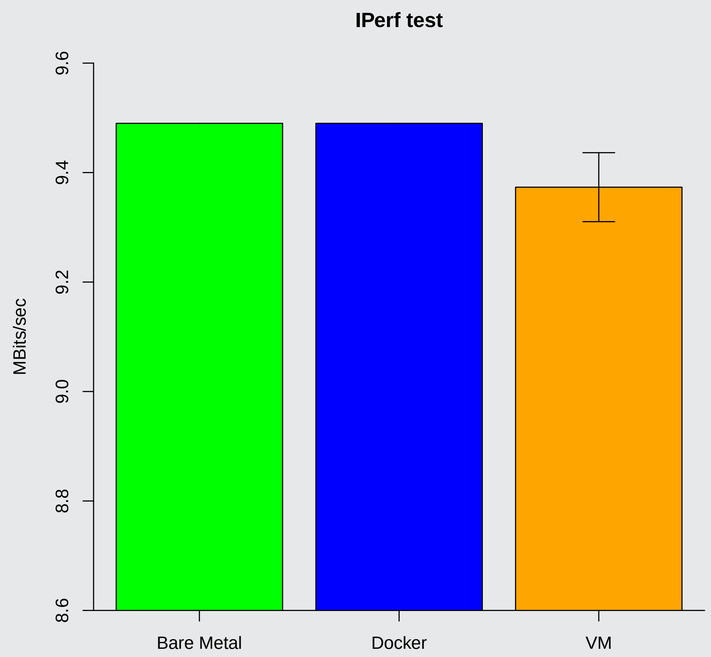
HEPSPEC allows us to quantify the performance of a CPU and hence see if any performance is lost due to containerisation or virtualisation.



The management overhead due to the hyper-visor is clear for the virtual machine where a significant drop in HEPSPEC score is seen. Containerisation shows a small drop of 1-2 HEPSPECS, but this is within the noise for Bare Metal.

6. IPerf

IPerf is used to test network performance and the effect containerisation or virtualisation has on this.



The IPerf plot shows that bandwidth is not affected by containerisation, however there is a small drop for virtualisation.

7. Conclusion

Virtualisation is shown to have significant although small overheads. These overheads are almost non-existant for containerisation. The most significant overhead in virtualisation comes from virtualising the disk, and this is obvious from the Bonnie++ results. Whilst no significant overheads are seen in containerisation, what has not been tested is the cumulative affect of these when multiple containers are running on a system.

Containerisation certainly shows promise as an alternative to full virtualisation, although it must be noted that certain cases of virtualisation cannot be containerised. These cases are mainly those where a different type of OS is required such as windows on a linux host.

http://www.hep.lancs.ac.uk/∼rel
r dot long at lancaster dot ac dot uk