**CMPSCI 577 Operating Systems Design and Implementation** **Spring 2020**

# Lecture 13: March 05, 2020

*Lecturer:* ***Prashant Shenoy*** *Scribe:* ***Wenjun Huang***

## 13.1 OS Virtualization

Last lecture we covered virtualization at the hardware level. This lecture we'll discuss virtualization at the OS level.

With OS level virtualization, processes run inside containers, which run on a host OS. Containers support resource allocation and isolation. Compared to hardware level virtualization (type 1/2), OS level virtualization is more lightweight because no actual OSs are run inside the containers–the containers share the host OS' kernel, thereby reducing the overhead.

OS level virtualization has some disadvantages as well. The isolation it provides is not as comprehensive as that provided by hardware level virtualization, so one faulty container could potentially bring down all other containers.

So how does containers work on Linux? Processes can be assigned to namespaces. Processes in a namespace can't see resources (e.g. files) not allocated to that namespace, and they would think they are the only processes in the system. Namespace management can be done through cgroups (what and how much resources can be used) and chroot.

## 13.2 Proportional Share Scheduling (Fair Share Scheduling)

Under this scheduling scheme, each process is assigned a weight ("share"), and each process is allocated resources in proportion to their shares. Fairness is achieved by reallocating unused resources (also in proportion to process shares).

However, starvation is still possible for certain variants of this scheduling algorithm. For instance, with the credit-based scheme, processes receive some credits periodically. Now suppose we have 2 processes, one of which is initially inactive. In this case, the active process will run freely with all the resources and it won't accumulate many credits. Meanwhile, the inactive process will accumulate lots of credits. Later, if the inactive process becomes active, it can starve the other process because it has more credits.

## 13.3 Docker and Linux Containers

Docker is a wrapper over Linux containers that makes using containers easier. Also, Docker containers are self-contained, meaning they package their own dependencies.

However, since Docker depends on Linux, running Docker on non-Linux systems means that a Linux kernel must be installed to support Docker. In such cases, the Linux kernel would run as a type 2 VM.

Docker uses a union file system called AuFS. A union file system is essentially a copy-on-write FS. If multiple containers read the same file, they would be accessing the same copy. When a container modifies a shared file, a copy is made to contain the changes. This is similar to how Git works. Such a FS allows Docker to reduce disk usage and produce small container images.

## 13.4 Virtual Machine Migration

Virtual machines can stay alive (i.e. no downtime) while being migrated between physical machines. State changes during migration can be handled by iteratively transferring memory.

One possibility is pre-copy VM migration. Under this scheme, we first send all memory pages to the destination. During the transfer, any changed memory pages will be marked dirty. Once the first transfer is completed, we transfer the dirty pages again. We repeat this process until the set of dirty pages is small enough. Then we pause the VM and transfer the remaining dirty pages to the destination. Afterwards, the VM may continue running at the destination.

Another possibility is post-copy VM migration. In this case, the non-memory states of the VM is transferred to the destination first. The VM then resumes running, and memory pages will be sent gradually to the destination in the background. Before the background transfer finishes, any needed pages can be copied on demand through asynchronous page fault.

Network connections to the VM will not be broken because IP addresses are bound to the VM's network interfaces, which won't change. The MAC addresses, however, are bound to the physical machine's network interfaces, so after migration, the VM needs to send an ARP packet to the gateway/switch to update its MAC-IP mappings.