

50004 - Operating Systems - Lecture 19

Oliver Killane

30/12/21

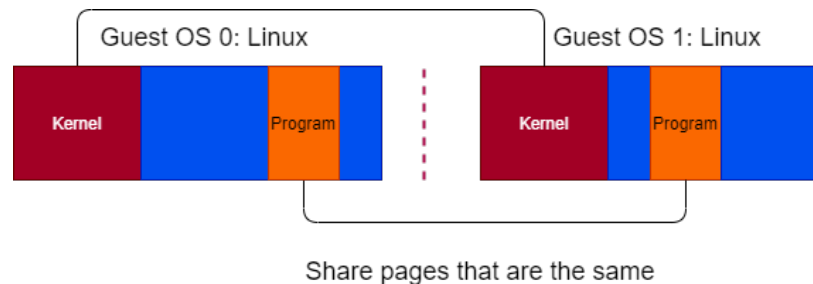
Memory Virtualization Continued...

Balloon Driver

Instead of relying on the **VMM** to determine pages to evict, force the Guest OS to evict pages.

- **Balloon Driver** installed in OS.
- Memory allocated to the balloon is available to the **VMM**.
- When the balloon is *inflated*, memory is reclaimed by **VMM** as low memory prompts the Guest OS to evict pages.
- When balloon is *deflated*, more memory is available to Guest OS.

Sharing Memory between VMs



By sharing identical pages (e.g for kernel, program information when VMs are on the same OS), memory usage is reduced.

Comparing pairs of pages would be too expensive, instead contents of pages is hashed. Only pages with identical hashes are compared, and if identical, shared.

There are many possible policies for when, and how to determine which pages to share. For example VMWare's ESX attempts to share pages when paging out to disk, and scans guest pages randomly at a predefined rate.

Containers

Processes are already isolated in terms of:

- Non-privileged CPU instructions & registers
- Virtual Memory
- File System
- System Calls (e.g for I/O)
- Signals

To improve this isolation between processes:

- **Support for Namespaces** e.g process identifiers are allocated and unique in groups, rather than process identifiers being global.
- **Separate Root Filesystems** Split the filesystem into many smaller, separate filesystems. Processes can create their own **jails** to ensure file system isolation. Useful for security (e.g split a companies internal and external services through jails), with very little overhead.

This is the main motivation behind containers.

