QoS-Aware Virtualization

Christopher Feener
Advisor: Xuechen Zhang

WASHINGTON STATE UNIVERSITY VANCOUVER

Motivations

- Want faster system with limited swapping.

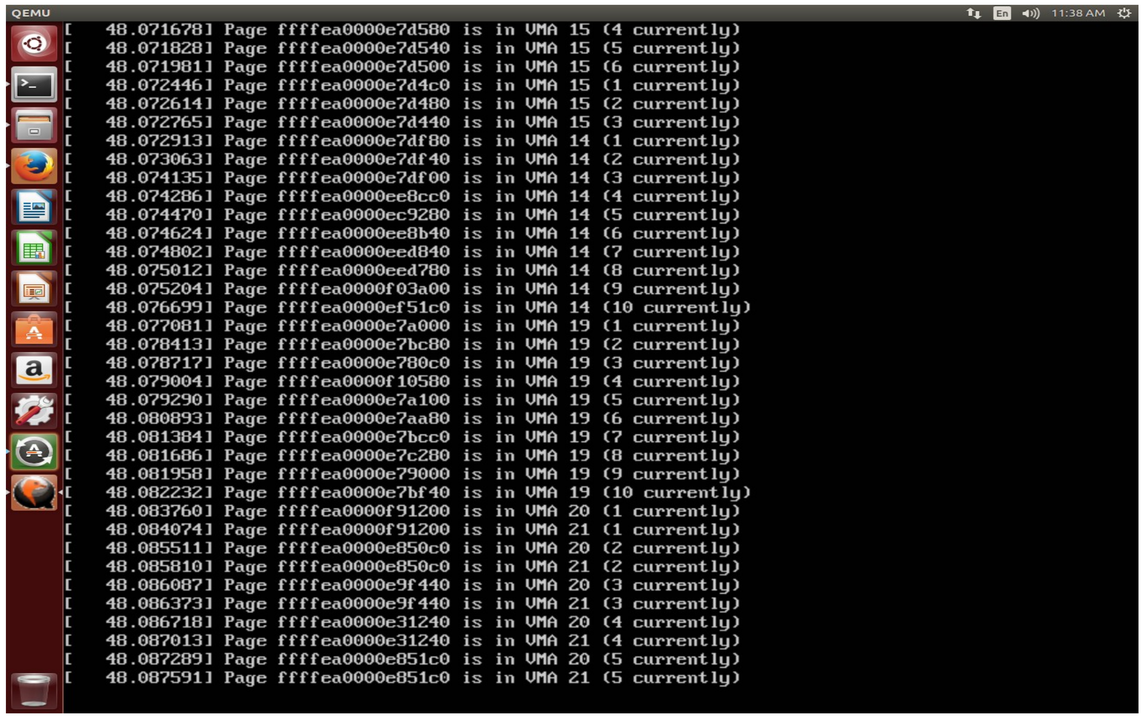
Challenges

- Speed of SSD is fast, but not durable.
- Disk is durable, but slow.
- Last attempt of “Hybrid Swapping” was done on Linux kernel 2.6, and many parts of the kernel have since changed.
- Lack of documentation.

Our Approach

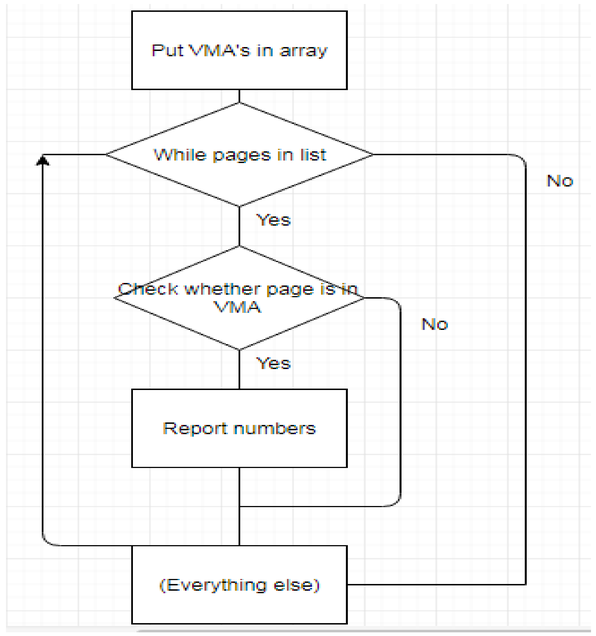
- “Hybrid Swapping” could use the advantages of both speed and limited swapping. It did this by swapping only page faults to disk.
- Only minor additions were made to linux kernel 4.11, specifically in the function *shrink_page_list()*.
- Debugging: No major program/method was used other than dmesg and printk statements.

Results and Data



Immediate Goal: Spatial Locality

- The “Hybrid Swap” method was meant to use both spatial and temporal locality.
- Spatial locality deals with how close addresses from a sequence are.
- By iterating through processes from the current task, virtual memory areas (VMA's) were put in an array.
- As the function iterated through pages, each page was checked to all the VMA's, using *page_address_in_vma()*.
- Each “match” had its information printed.
- These matches could be saved and recorded in another array for future use, such as analyzing the average distance between page addresses.



Setup

- Started with Qemu program, booting from kernel source code inside.
- Tested with ImageMagick on a large image of a galaxy. This forced the computer to have page faults and use swap.
- Coded inside function *shrink_page_list()*.

Long-Term:

Conclusion

- First
- Second