A Case for Page Table Reclaim

Karim Manaouil & Antonio Barbalace
The University of Edinburgh

Content

- Virtual Memory
- Cases for Reclaim
  - Sparse Mappings
  - Virtualisation
  - Memory Hotplug
  - TLB Stalls
- Reclaiming Page Tables in the Linux Kernel
- Observations
- Ongoing work
- Conclusion

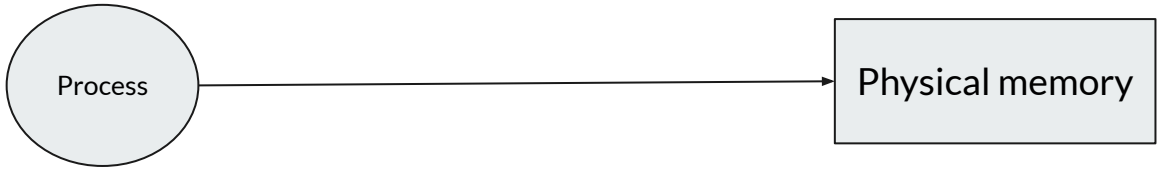
Virtual Memory: The pillar of memory management

1. Isolation and memory protection between processes
2. Gives super powers to the operating system kernel
    2.1. Allocating memory on demand (a.k.a demand paging)
    2.2. Swapping (to manage overcommitment)
    2.3. Transparently mapping memory from different devices (DRAM, CUDA/GPU, NVRAM)
    2.4. Copy-On-Write (CoW) for fork
    2.5. Various advanced memory management features
        2.5.1. NUMA balancing
        2.5.2. Huge pages
        2.5.3. Virtualisation
        2.5.4. Page cache

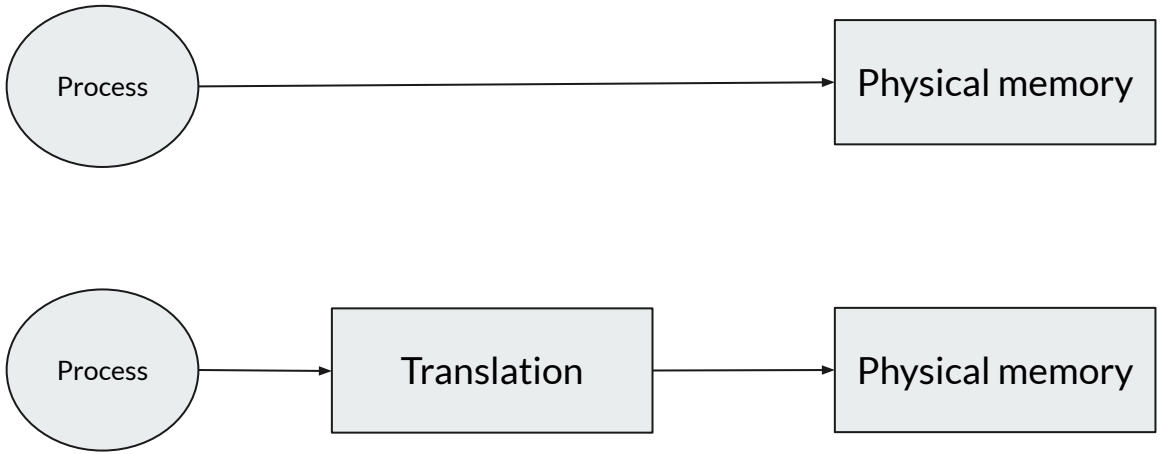
Virtual Memory & Page Tables

“All problems in computer science can be solved by another level of indirection.” - David Wheeler

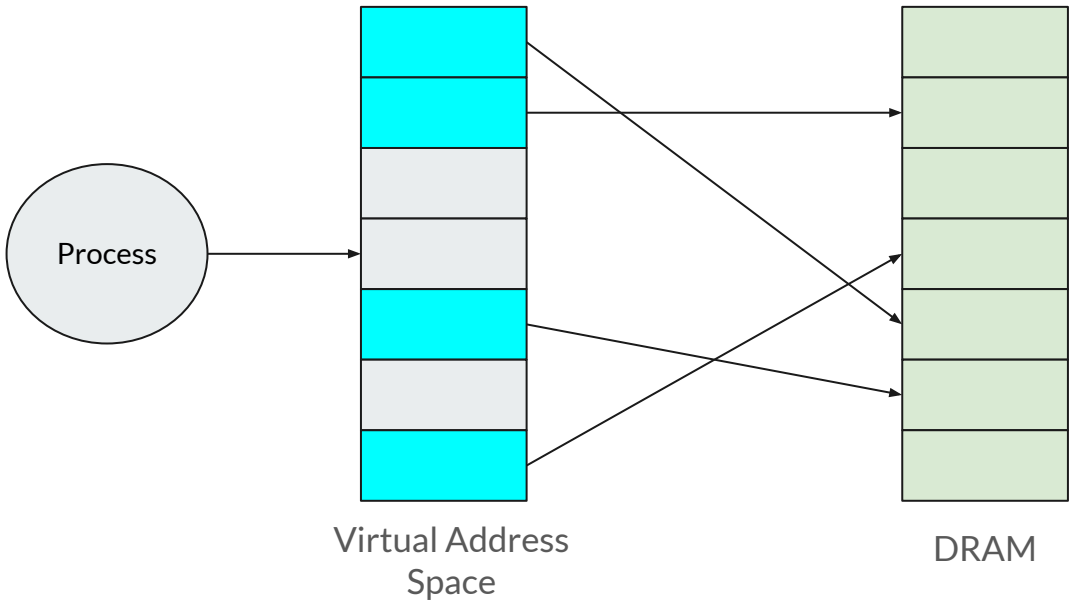
Virtual Memory & Page Tables



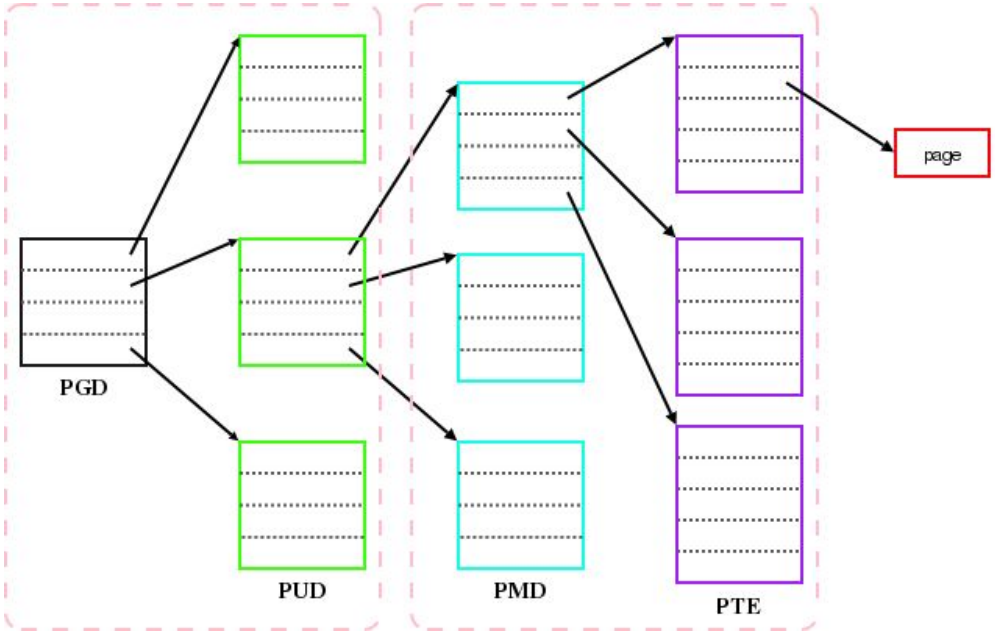
Virtual Memory & Page Tables



Virtual Memory & Page Tables



Virtual Memory & Page Tables



Page Table Math

- PTEs can map up to 2MiB of memory
- Each PTE is 4 KiB
- 64 GiB working set requires (64G / 2M * 4K) = 128 MiB
- In general, for X GiB, 2X MiB is required

The Problem of Sparse Mappings

- Libraries like Jemalloc/Tcmalloc leverage sparse mappings for various features
  - Fragmentation avoidance
  - Huge-page aware allocation

- Create a lot of virtual maps over the course of execution
- Call `madvise(MADV_FREE)` to free the physical pages, but not the maps
  - To avoid holding mmap locks
  - Avoid rebuilding VMA and page tables

- Page tables accumulate over time and stay unreclaimed

The Problem of Sparse Mappings

- Redis uses jemalloc by default
- On a long-running 600GiB instance of Redis, 100 GiB of physical memory was page tables

The Problem of Memory Hotplug

- Memory desagregation (physical memory allocation on demand) relies on hotplug
- Hotplug is the process of adding/removing memory from a running system
- Memory ballooning also extensively relies on that
- Page tables are unmovable allocations
  - Network buffers
  - Kernel objects
  - Page tables

- Hotunplug fails with unmovable allocations

The Problem of TLB Stalls

- Linux opportunistically uses huge pages with THP (Transparent Huge Page)
  - Reduce TLB stalls
  - Improve access latency
- THP requires allocating higher-order pages (e.g. 2 MiB)
- Unmovable allocations hinders compaction
- Page tables can increase TLB stalls

Reclaiming Page Tables in Linux

- Page tables with no valid references
- Unutilised page tables under memory pressure
  - Host native page tables
  - Guest second-level page tables

Unreferenced Page Tables

- Add a per-cpu reference counter to the page structure
- Whenever an entry is mapped, the ref counter is atomically incremented
- Whenever an entry is unmapped, the refconter is atomically decremented
- Once it reaches zero, the refcount is freezed
    - Concurrent threads won't get a ref
- PMD entry is cleared and page table is freed

PTE Reclaim under Memory Pressure

- Second-level PTEs for VMs always contain valid entries
- If the system is short on memory, page tables can be reclaimed
- All PTEs are inserted into a per-memcgroup list upon allocation
- Memory pressure randomly picks PTEs from the list
  - Freeze the refcount (to prevent concurrent reclaim)
  - Empty and zero page references only are freed
  - PTEs with valid references are
    - Migrated to another NUMA node
    - Compressed in-memory
    - Swapped out to disk

Observations

- On a microbenchmark with madvise(), went from 100 MiB PTEs to 100 KiB
- No overhead for refcount reclaim
- Swapping helps manage memory overcommitment situations
    - Reduced OOM killer invocations

Ongoing/Future work

- It is only applied to anonymous mapping for now
  - Page cache on the way
- PTE list is not LRU sorted
  - Need a mechanism to track access frequency of page tables
- Leverage migration for compaction
- How much does it help for reducing TLB stalls?

The End

Thanks!

Questions?