The Story of a Page

Rahul Iyer

Agenda

- The Page Descriptor
- The Various Page Lists
- The Evictor Interface
  - Kswapd
  - The Direct Reclaim Path
- The Life Cycle of a Page
- Conclusion

The Page Descriptor

- Defines a *physical* page
- One for every physical page in memory
- All global descriptors stored in the array `mem_map[]`
- Two ‘list’ fields
  - `struct list_head list`
  - `struct list_head lru`

I’m Lost… Where am I?

- Page can be part of 2 lists
- Through the `list` field
  - Example: Inode cache
  - Not of real interest to us
- Through the `lru` field
  - Part of the active/inactive list
  - Part of the slab allocator

The Evictor Interface

- The evictor basically has two parts to it
  - The kswapd ‘page out’ daemon
  - The Direct reclaim Path

The Page Out Daemon

- The kernel thread – kswapd
- Executes the function `kswapd()`
- Created at startup
  - Created by `kswapd_init()`
  - `kswapd_init()` is `__init()`
- Sleeps for the most part
- Woken up in cases of low memory
- Primary call path is `balance_pgdat()`

The Page Out Daemon (Contd.)

- `balance_pgdat()` calls
  - `shrink_zone()`
  - `shrink_slab()`
- `shrink_zone()`
  - is the per zone page freer
  - Used by kswapd and Direct Reclaim
- `shrink_slab()`
  - Shrinks the slab caches in the slab allocator
  - Used by both kswapd and Direct Reclaim

The Direct Reclaim Path

- This is when process try to free memory directly – the situation is desperate
- Bypass kswapd
- The main entry is `try_to_free_pages()`
- `try_to_free_pages` calls
  - `shrink_caches()` calls
    - `shrink_zone()`
  - `shrink_slab()`
- May involve a call to the OOM killer

Life Cycle of a Page

- Page allocated using `alloc_pages()`
- Added to the inactive list or Slab cache
- On access, marked accessed by `mark_page_accessed()`
- Moved back to inactive list by `refill_inactive()`
- In low memory situations, reaped by kswapd or Direct Reclaim

Conclusion

- The evictor has two ‘entry’ points
- Deals closely with the Page cache
  - Means that Page cache structure will be altered
  - Hairy as the page cache is a cache of used pages + Buffer Cache
- Next Milestone – figure our the call sites of all page cache functions