Xen Self-Ballooning

Memory Overcommitment Management in SQL

Transcendent Memory

Intro

- Allows RAM to be shared across kernels.
- "The end goal is that memory can be more efficiently utilized by one kernel and/or load-balanced between multiple kernels".
- Implementation is divided into into front-end and back-end. Front-end implementations are `frontswap` for anonymous pages and `cleancache` for file backed pages.

Frontends

- Frontends use the APIs/ABIs provided by the tmem backend.
- They work independent of the backend implementation.
- Currently there are two frontends in the linux kernel - `frontswap` for the anonymous pages and `cleancache` for mapped pages.

Frontswap

- Allows the Linux swap subsystem to use transcendent memory in place of the swap device.
- When a page is to be swapped, it is first sent to tmem.
- If tmem accepts it, then it can guaranteedly be retrieved back from it at a later point of time.
- If tmem rejects it, then the page is written to the normal swap device.

Cleancache

- Mapped pages can be reclaimed by the kernel any time. If the same page is to be used again, page fault happens and it is fetched from the disk
- When a page is to be swapped, it is first sent to tmem. If tmem accepts it, fine, else the page is removed as usual.
- Cleancache data in tmem is ephemeral, which means that a page can be discarded if tmem chooses, no guarantees like frontswap provided.
- When kernel needs that page back, it asks tmem. If tmem has retained the page, it gives it back; otherwise, the kernel proceeds with the refault, fetching the data from the disk as usual

Backends

- Backends for tmem include `zcache` and `Xen tmem`.
- Xen tmem is for virtualized environment, while zcache can be used in non-virtualized environments.
- A new backend called RAMster is under development.

Zcache

- Compresses the pages before storing, so more pages can be stored
- When the kernel has N pages to store but available memory to store them is less than N*pagesize, it can put them in zcache.
- It usually rejects pages which have a low compression factor to avoid storing poorly compressible data

Xen Transcendent Memory

- The spare memory of the hypervisor is used as tmem.
- All the guests first check if page can be stored in tmem before swapping them out.
- Implements both compression and deduplication (both within a guest and across guests) to maximize the volume of data that can be stored

Frontswap Self-Shrinking

- When kernel swaps a page, it assumes that the page will go to disk and may remain there for long time even if it is not used again
- Kernel assumes disk space is less costly and abundant
- A page in frontswap may be taking up valuable space that is needed for some other purpose.
- To resolve this, there is frontswap self shrinking, which creates a partial swapoff condition, which prompts the kernel to retrieve pages from swap.
- When a guest is under normal memory pressure, this reclaims pages from tmem.

Self-Ballooning

- A self-ballooning driver is present in the guest OS. Requires tmem to be enabled.
- A process runs regularly after a fixed time interval(configurable) which sets the target size.
- Target Size = Committed pages + reserved pages + balloon reserved pages (used to reserve pages for page cache etc. with default value = 10% of the total ram).

- If the target is less than current ram, guest is ballooned down, else it is ballooned up.
- There is a hysteresis counter which represents the number of iterations it will take for machine to balloon down to target
- So, each time self-ballooning process runs, RAM = current - (current-target) /hysteresis_counter
- a `min usable mb` parameter specifies the minimum amount of ram the guest should have. Machine cannot be ballooned down below this point.

Conclusion

- Tmem is implemented in Xen, while implementation for KVM is not present.
- Xen tmem does not seem to be useful during memory crunch on the host, but when spare memory is available in a physical machine, then it speeds up the performance of the machines.

References

- https://lwn.net/Articles/454795/
- Code for xen self-balloon driver present at https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/drivers/xen/xen-selfballoon.c?id=refs/tags/v4.4-rc7