



Contribution ID: 209

Type: **not specified**

Transitioning `get_user_pages` (GUP) to folio based mapping

Friday 20 September 2024 16:15 (15 minutes)

Direct and passthrough IO involves mapping user space memory into the kernel. At present, this memory is mapped as an array of pages. Using 4K pages for mapping results in additional overhead due to per-page memory pinning, unpinning, and calculations. Switching to a large folio-based mapping will reduce this overhead.

As part of this proposal, the current GUP implementation needs to be updated to use folios. This change must address the following aspects:

1. The page table walk should check for larger folios and fetch a bigger contiguous folio. This will also handle Copy-On-Write (COW) situations that might modify otherwise contiguous pages within a large folio.
2. The page table walk of GUP deals with huge pages that currently return an array of pages. The huge page helper functions should be modified to fetch larger folios.
3. The GUP slow path, currently based on a pages array, should be converted to return a collection of folios.
4. GUP processing based on `folio_batch` will also avoid the need for allocation of page/folio arrays.
5. There are many users of GUP functions that operate on array of pages. Should both `get_user_pages` and `get_user_folios` exist, or just the latter?

Primary author: KUMAR, Kundan (Samsung Semiconductor India Research)

Presenter: KUMAR, Kundan (Samsung Semiconductor India Research)

Session Classification: Kernel Memory Management MC

Track Classification: Kernel Memory Management MC