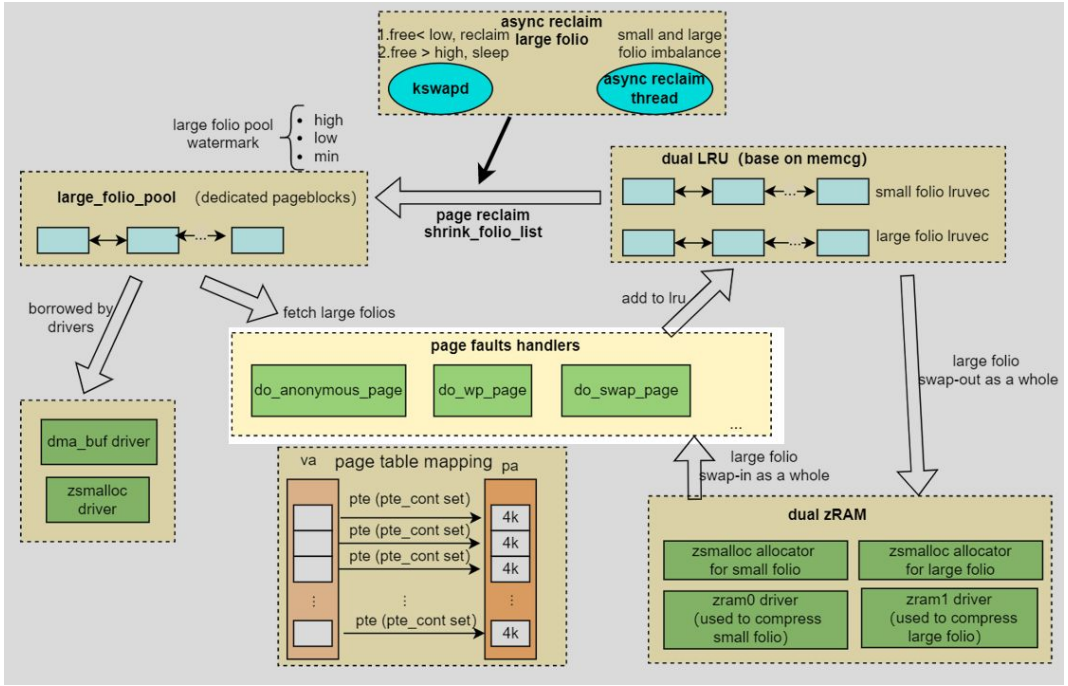
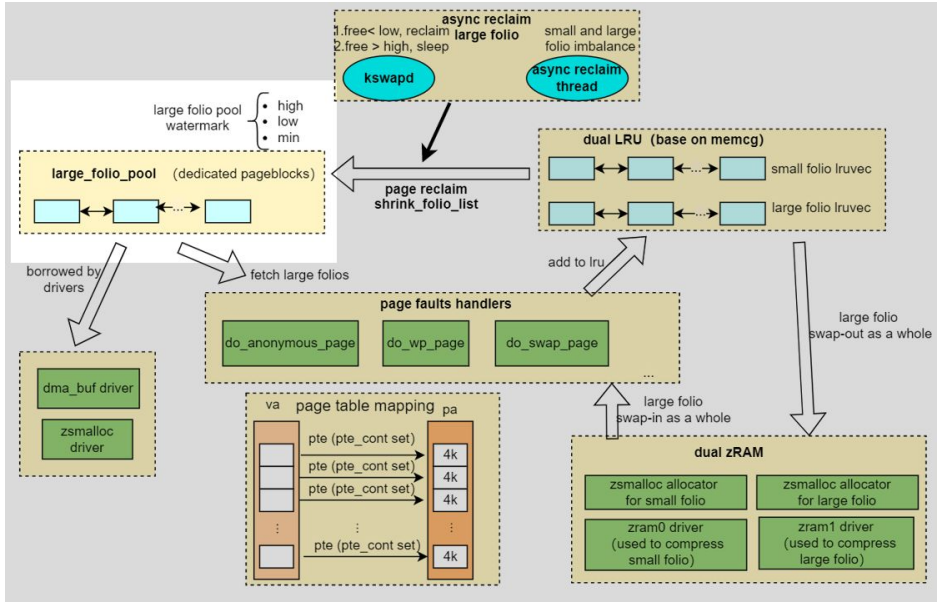
Product Practices of Large Folios on Millions of OPPO Android Phones

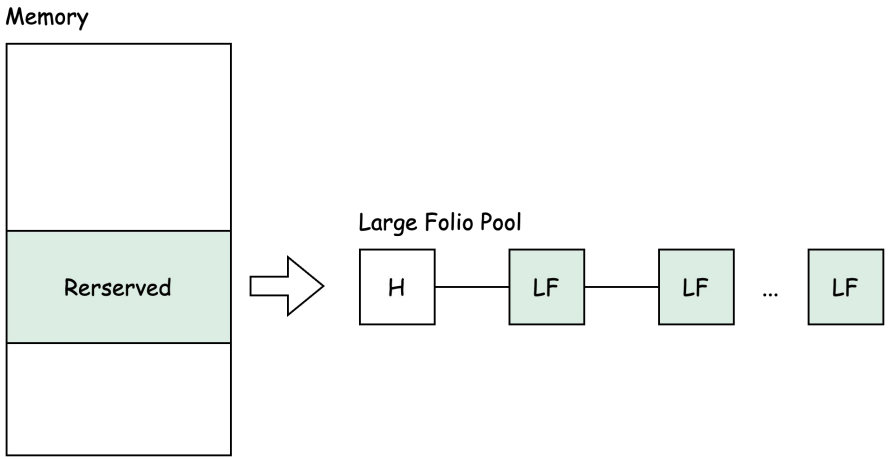
Software Architecture - Page Faults

Try large folio at first, if it fails, use small folio

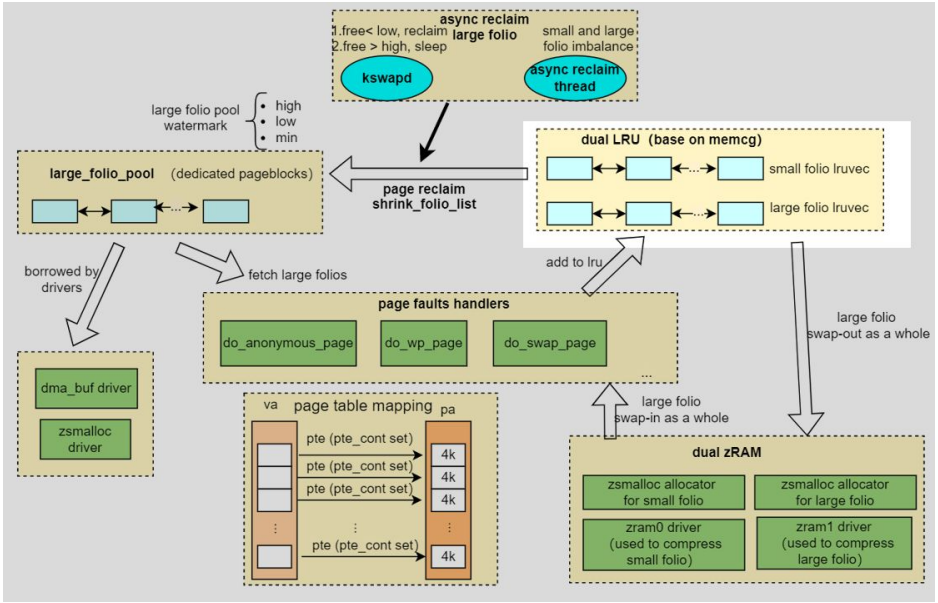


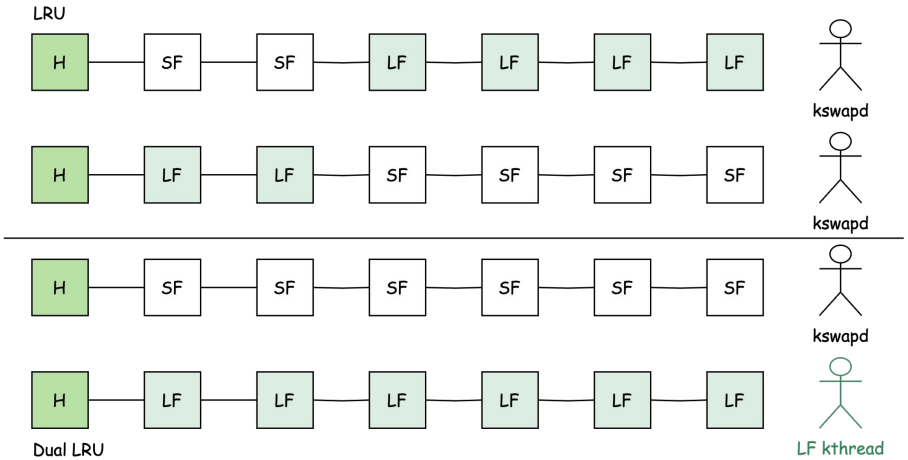
Software Architecture - Large Folio Allocator



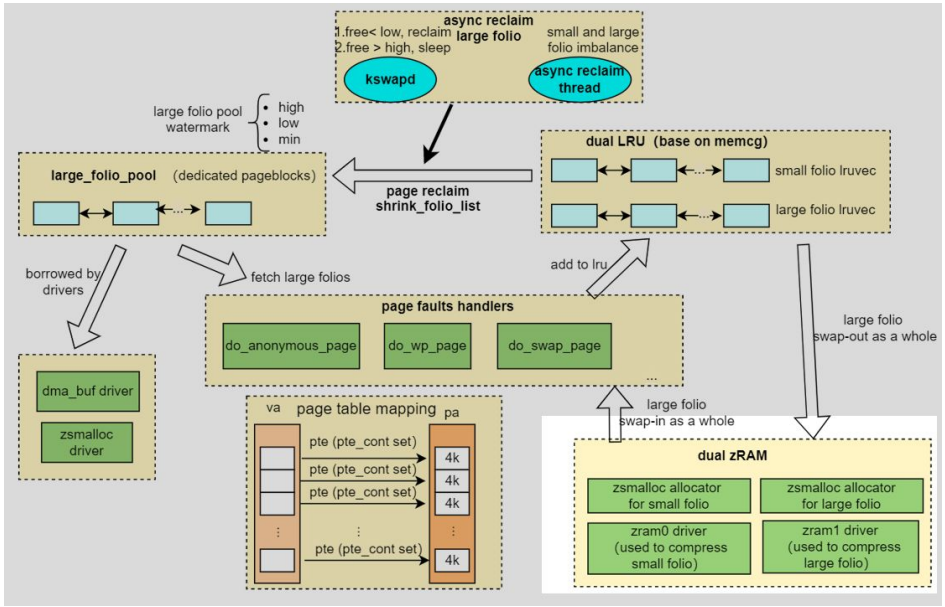


Software Architecture - Dual LRU

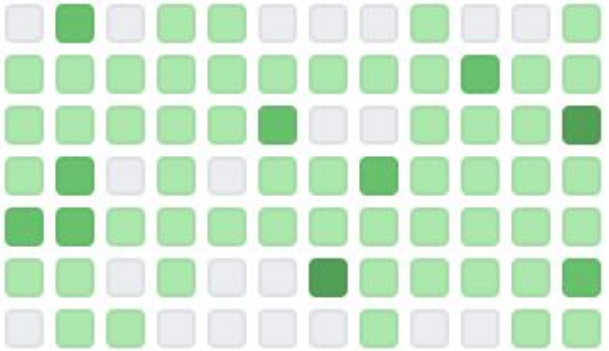




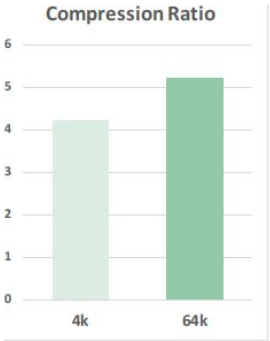
Software Architecture - Dual Zram

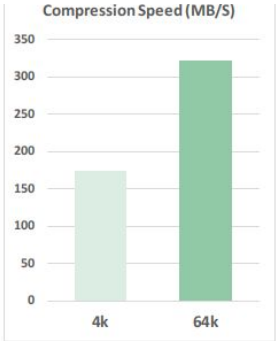


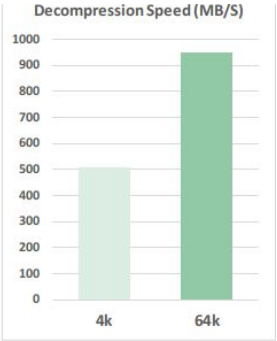
Swap Fragmentation



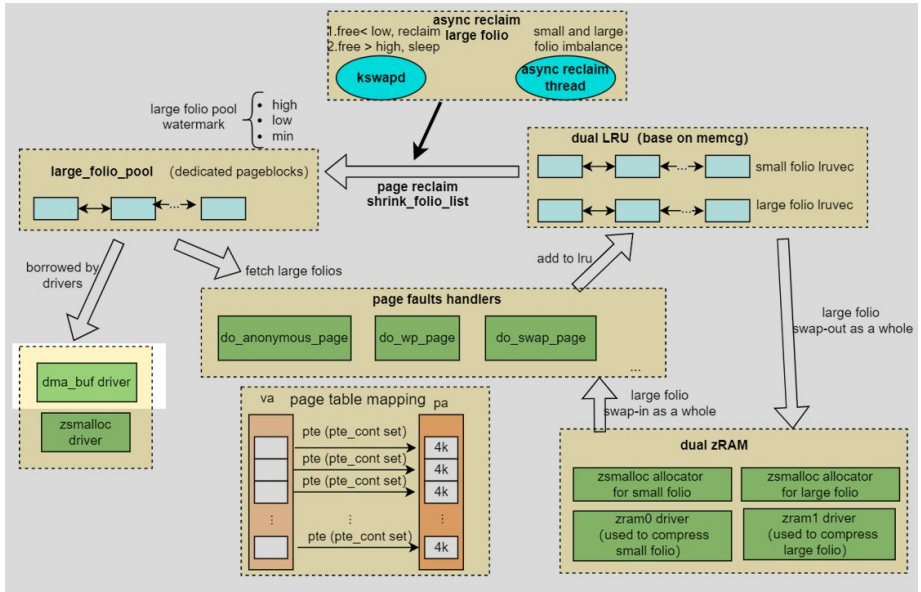
Performance Testing of 64k and 4k Folio with zstd Compression Algorithm







Software Architecture - System Heap



android-mainline > common/drivers/dma-buf/heaps/system_heap.c

system_heap.c

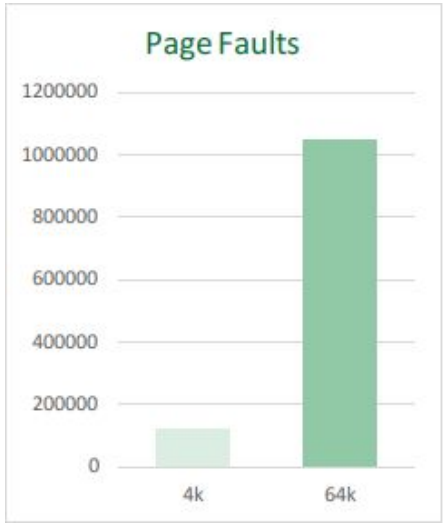
```
51 #define LOW_ORDER_GFP (GFP_HIGHUSER | __GFP_ZERO)
52 #define HIGH_ORDER_GFP  (((GFP_HIGHUSER | __GFP_ZERO | __GFP_NOWARN \
53                                 | __GFP_NORETRY) & ~__GFP_RECLAIM) \
54                                 | __GFP_COMP)
55 static gfp_t order_flags[] = {HIGH_ORDER_GFP, HIGH_ORDER_GFP, LOW_ORDER_GFP};
56 /*
57  * The selection of the orders used for allocation (1MB, 64K, 4K) is designed
58  * to match with the sizes often found in IOMMUs. Using order 4 pages instead
59  * of order 0 pages can significantly improve the performance of many IOMMUs
60  * by reducing TLB pressure and time spent updating page tables.
61  */
62 static const unsigned int orders[] = {8, 4, 0};
63 #define NUM_ORDERS ARRAY_SIZE(orders)
```

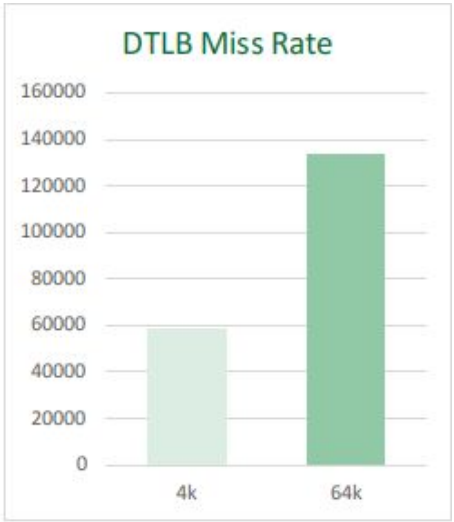
Use Large Folio

Performance Improvements

oppo

Significantly improves page faults and dtlb miss rate





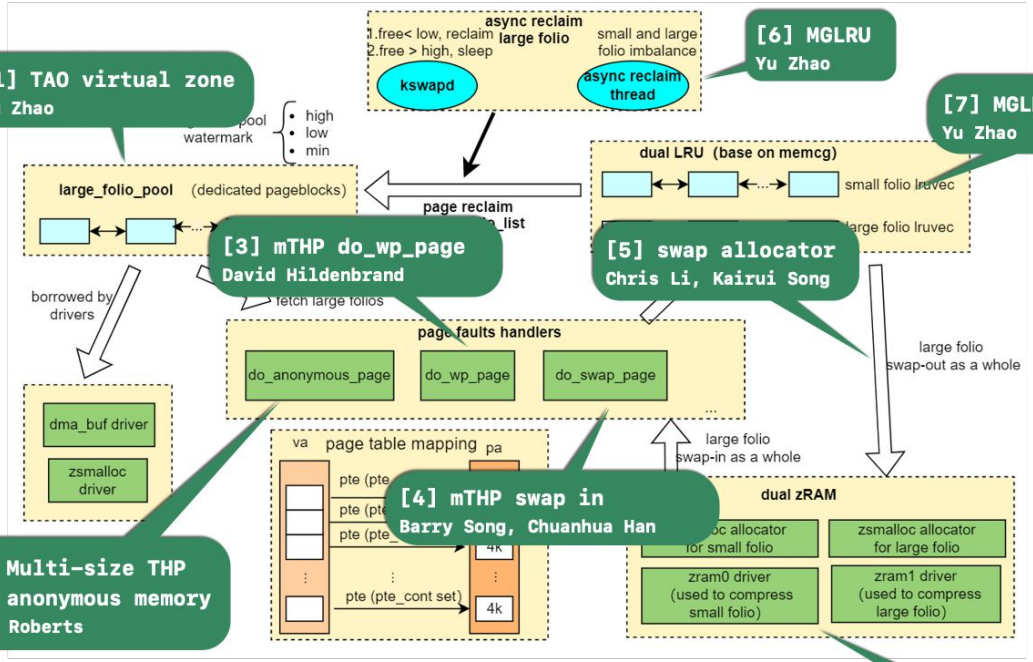
Benchmark

- **RamBench**
  Improved memory access performance by an average of **10%**
- **AndroBench**
  Database update performance improved by **16%**
- **AnTuTu Benchmark**
  Total score increased by **22,000+**

User Scenarios

- **App Launch Speed**
  Increased by **10%**
- **Frame Drop Rate**
  Decreased by **35%**

Upstream



[1] https://lore.kernel.org/linux-mm/20240229183436.4110845-1-yuzhao@google.com/

[2] https://lore.kernel.org/linux-mm/20231204102027.57185-1-ryan.roberts@arm.com/

[3] https://lore.kernel.org/all/20240829165627.2256514-1-david@redhat.com/T/#u

[4] https://lore.kernel.org/linux-mm/20240529082824.150954-1-21cnbao@gmail.com/

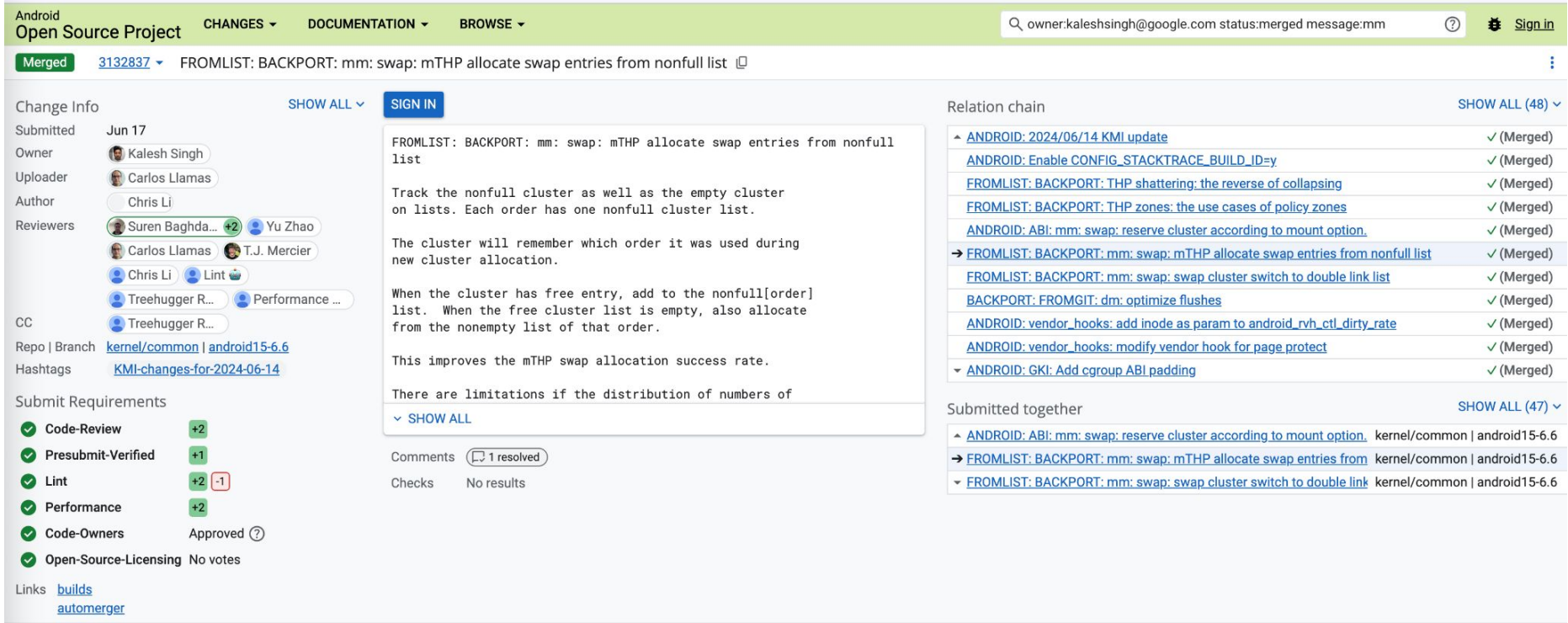
https://lore.kernel.org/linux-mm/20240908232119.2157-1-21cnbao@gmail.com/

[5] https://lore.kernel.org/linux-mm/20240524-swap-allocator-v1-0-47861b423b26@kernel.org

[6] [7] https://lore.kernel.org/lkml/20220706220022.968789-1-yuzhao@google.com/

[8] https://lore.kernel.org/linux-mm/20240327214816.31191-1-21cnbao@gmail.com/

GKI Backport



QA & Discussion

User Space:

- Native/Dalvik Allocator (mTHP awareness)

Kernel Space:

- FileSystem: F2FS/EROFS