# CS 5472 - Advanced Topics in Computer Security

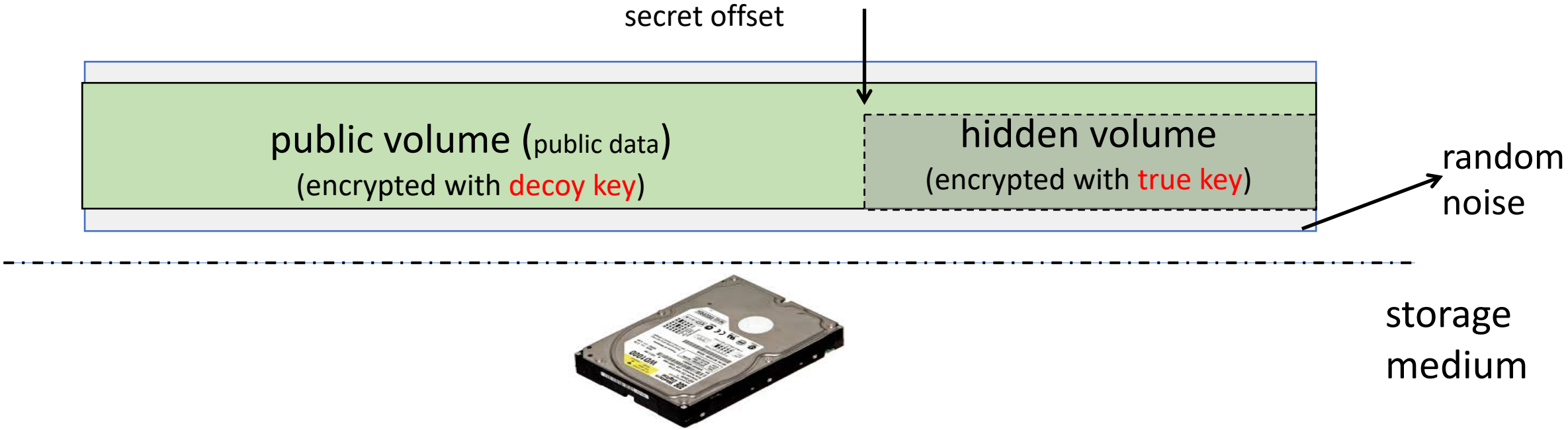
## Topic 5: Deniable Encryption (2)

Spring 2019 Semester
Instructor: Bo Chen
bchen@mtu.edu
https://cs.mtu.edu/~bchen
https://snp.cs.mtu.edu

# Review: Use Hidden Volume to Mitigate Coercive Attacks

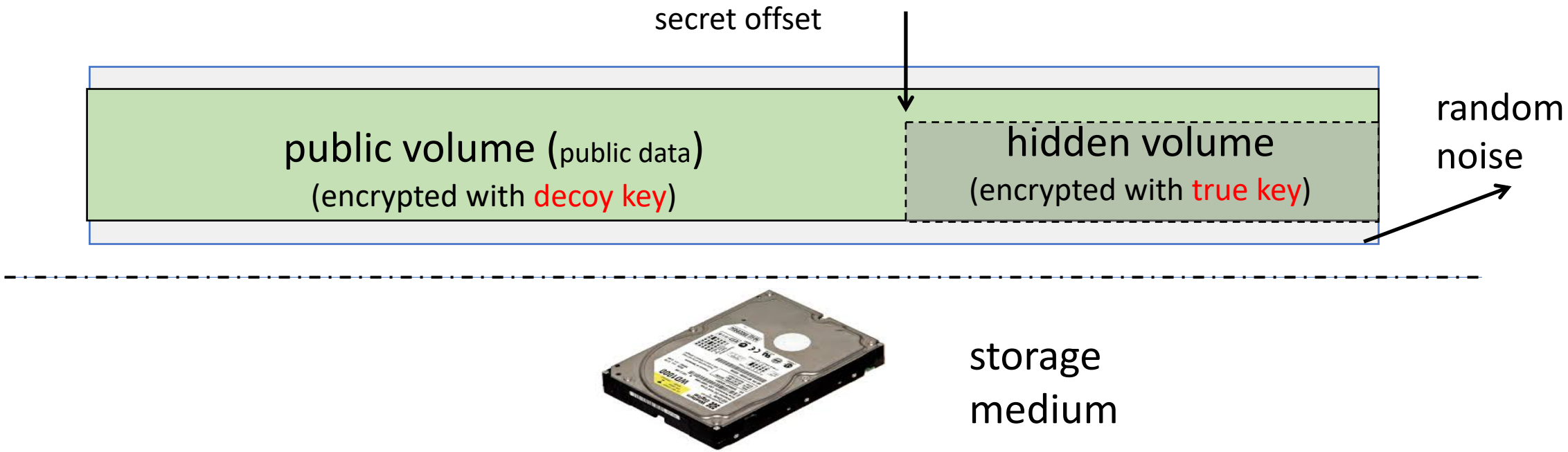
- A coercive attacker can enforce the victim to disclose the decryption key

- Mobiflage is a hidden volume-based PDE system which can mitigate coercive attacks for mobile devices

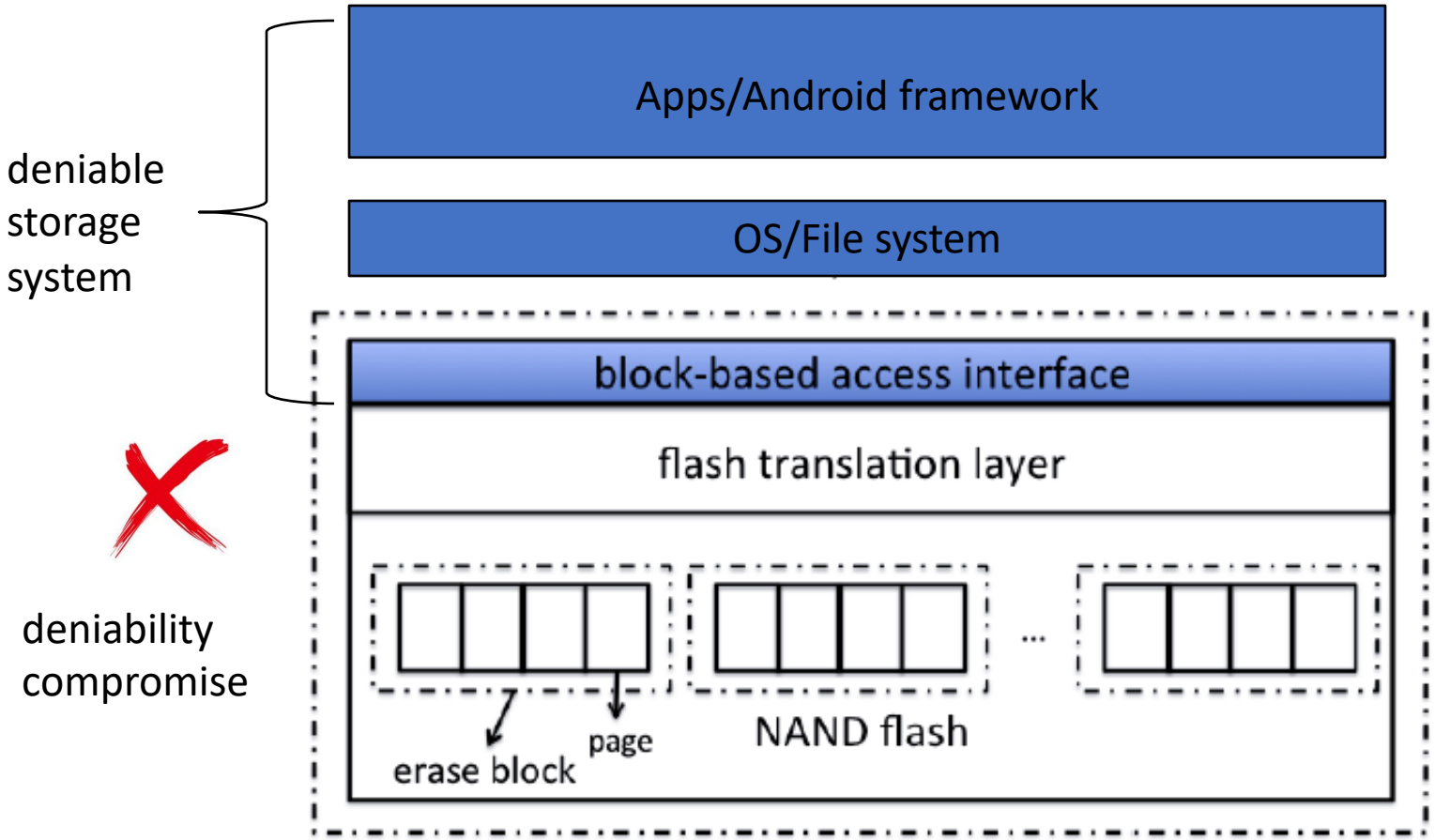



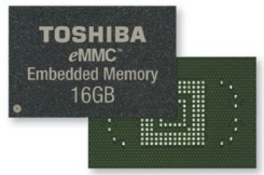
# Deniability Compromise 1: The Attacker Can Have Access to The Disk Multiple Times

- By having multiple snapshots on the storage medium, the attacker can compromise deniability
  - Compare different snapshots and can observe the changes/modifications over the hidden volume, which was not supposed to happen
  - **Hidden volume is hidden in the empty space of the public volume**



# Deniability Compromise 2: Underlying Flash Memory



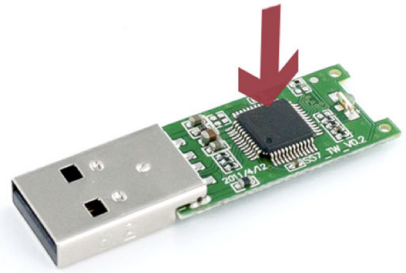

eMMC Chip

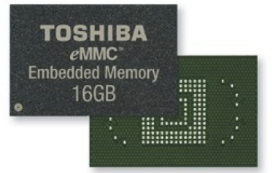
MMC Card


- Built on the block device layer
- Neglect the special nature of flash
- Unexpected ``traces'' of hidden sensitive data due to handling the special nature of flash memory

# NAND Flash is Usually Used as Storage Media

- NAND flash
  - USB sticks
  - Solid state drives (SSD)
  - SD/miniSD/microSD/eMMC

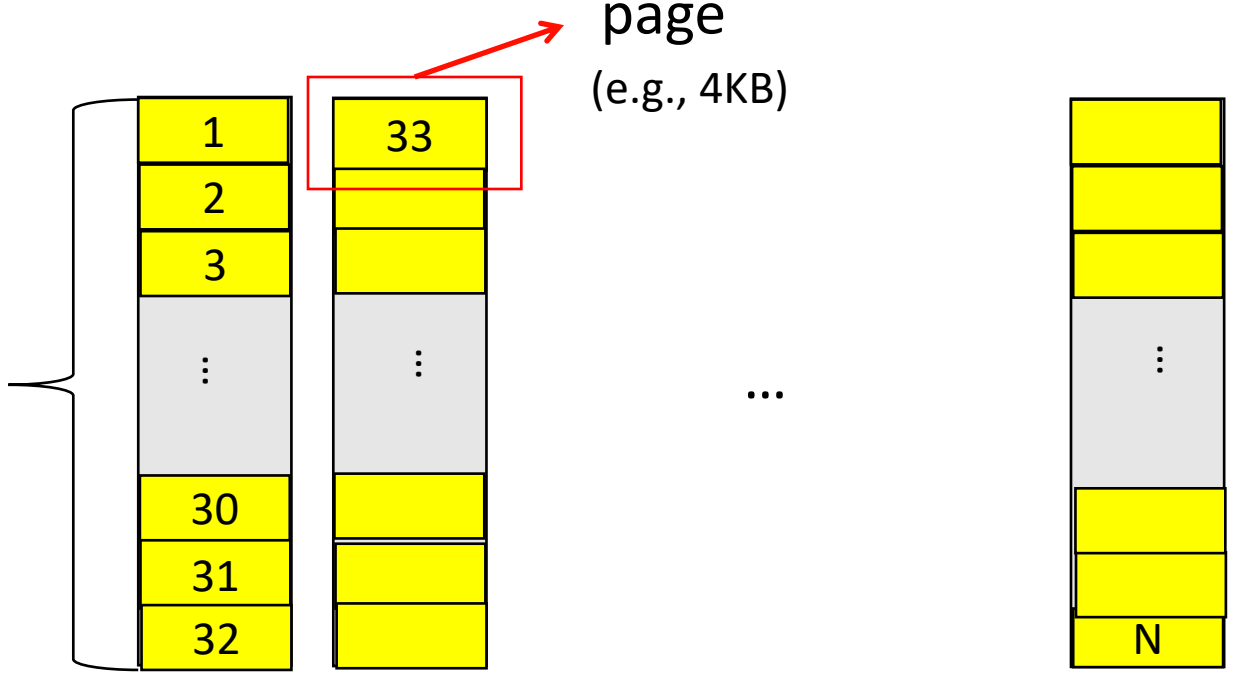





eMMC Chip

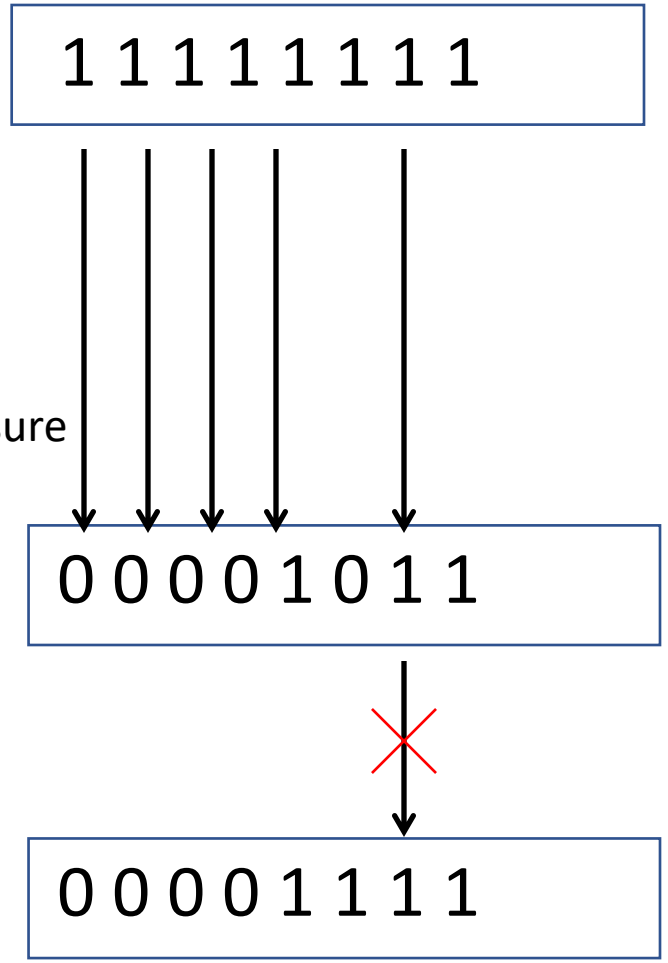
MMC Card





# How to Program/Write Data to Flash Memory

All ‘1’ initially:



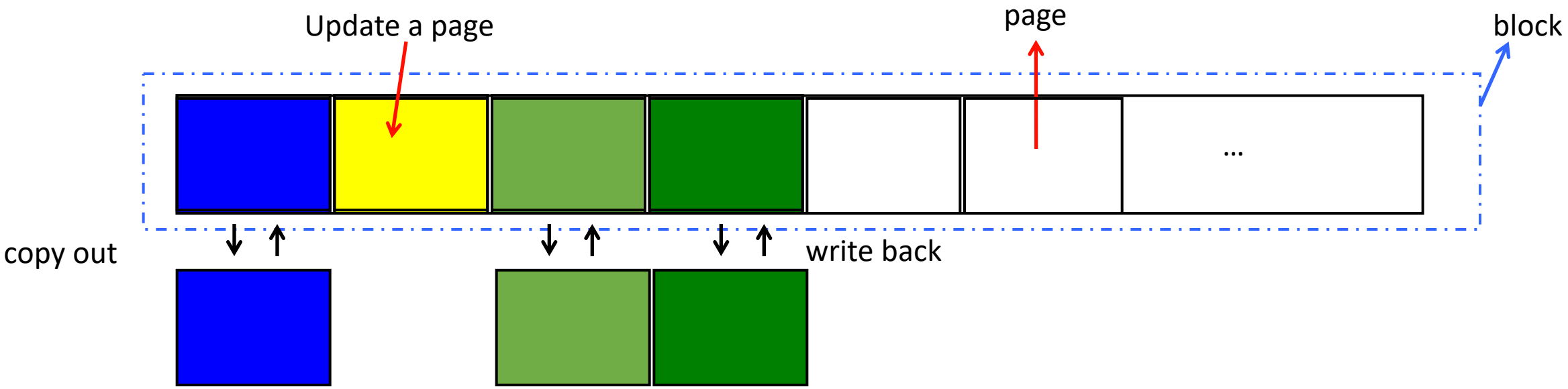

Write 0x0b:
Rule:
1) 1 can be programed to 0
2) 0 cannot be programed to 1 except performing an erasure

Write 0x0f?

Need to erase to all ‘1’ first

# Special Characteristics of NAND Flash

- Update unfriendly
  - Over-writing a page requires first erasing the entire block (erase-before-write)
  - Write is performed on basis of pages (e.g., 4KB), but erase is performed on basis of blocks (e.g., 128KB)




  - Over-write may cause significant write amplification

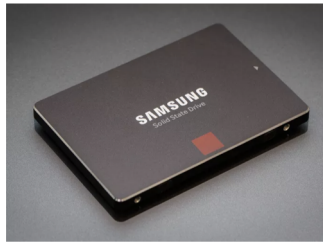
# Special Characteristics of NAND Flash (cont.)

- Support a finite number of program-erase (P/E) cycles
  - Each flash block can only be programmed/erased for a limited number of times (e.g., 10K)
  - Data should be placed evenly across flash (wear leveling)

# How to Manage NAND Flash?

- Flash-specific file systems, which can handle the special characteristics of NDND flash
  - YAFFS/YAFFS2, UBIFS, F2FS, JFFS/JFFS2

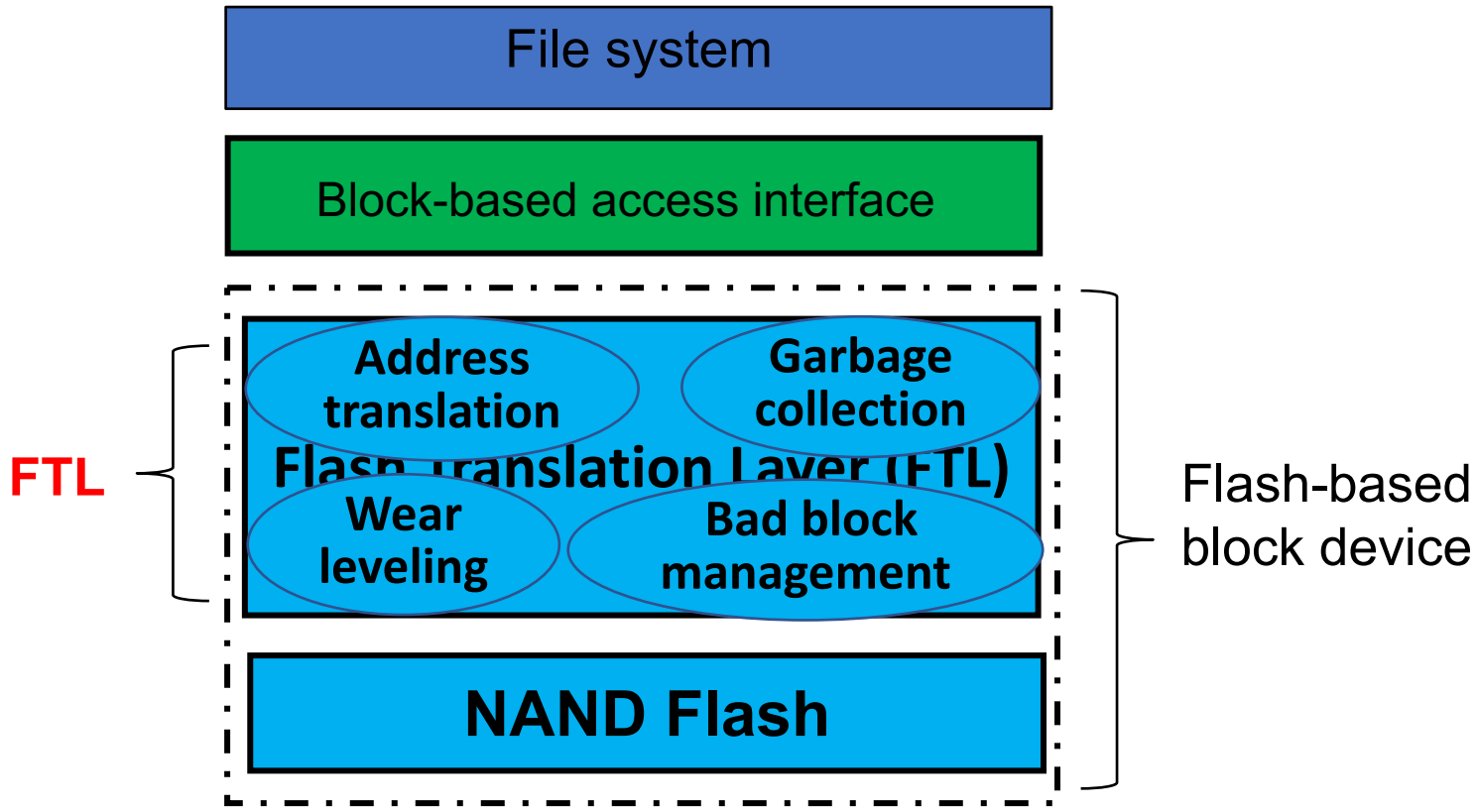
- Flash translation layer (FTL) – flash firmware embedded into the flash storage device, which can handle the special characteristics of NAND flash and emulate the flash storage as a regular block device
  - SSD
  - USB
  - SD
  - MMC




# Flash Translation Layer (FTL)

FTL usually provides the following functionality:

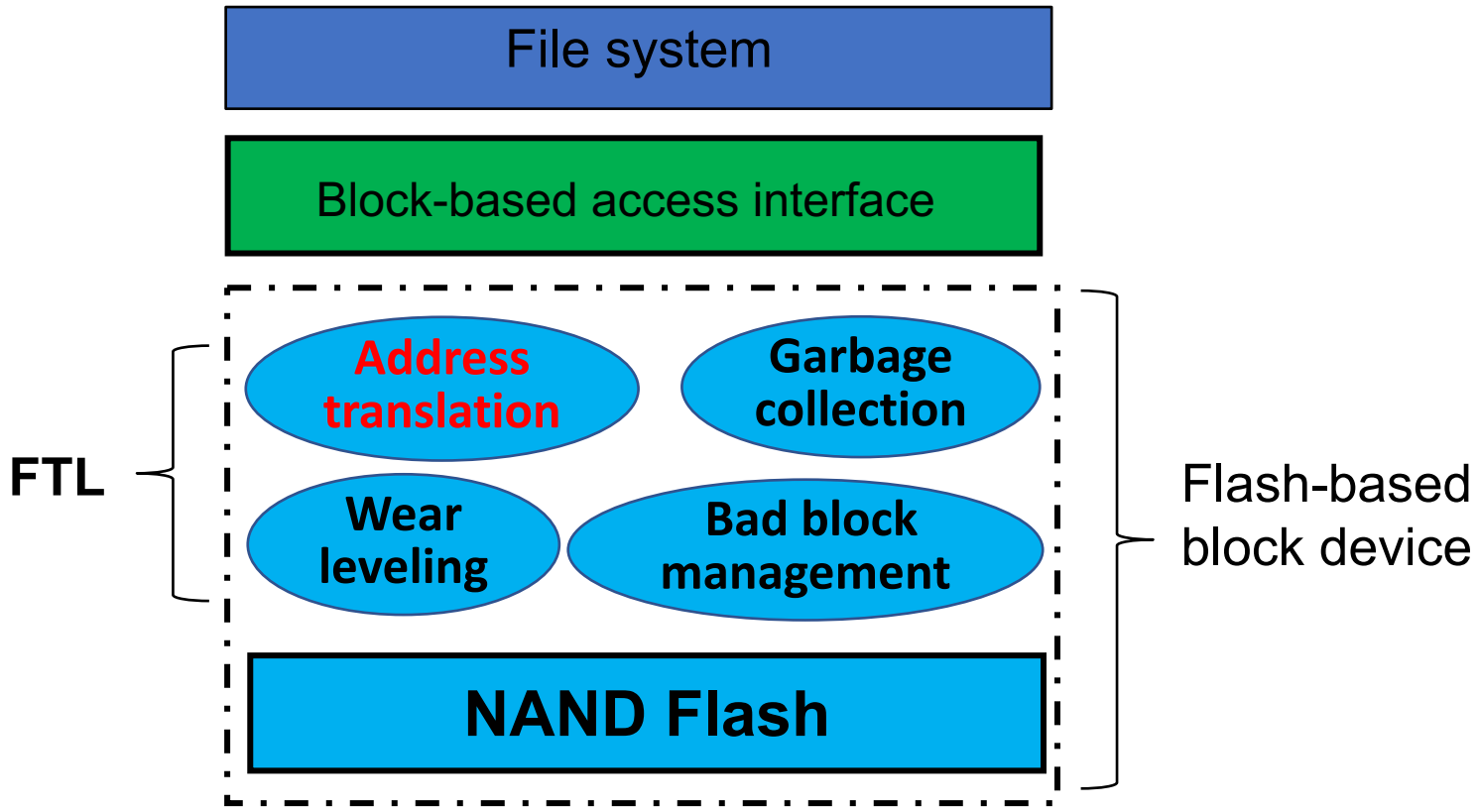
- ✓Address translation
- ✓Garbage collection
- ✓Wear leveling
- ✓Bad block management



# Flash Translation Layer (cont.)

Address translation:

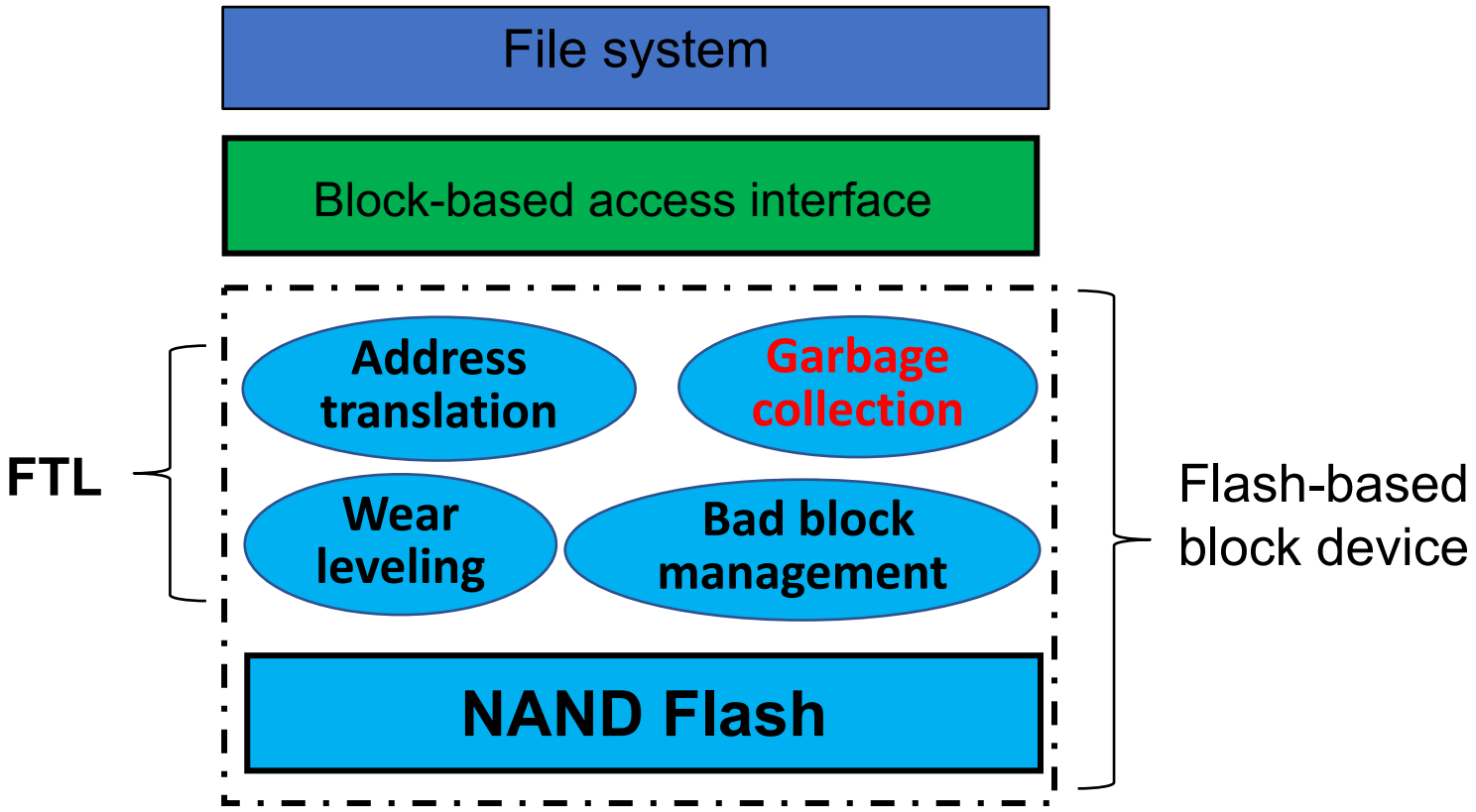
- ✓Translate address between block addresses and flash memory addresses
- ✓Need to keep track of mappings between Logical Block Address (LBA) and Physical Block Address (PBA)



# Flash Translation Layer (cont.)

Garbage collection:

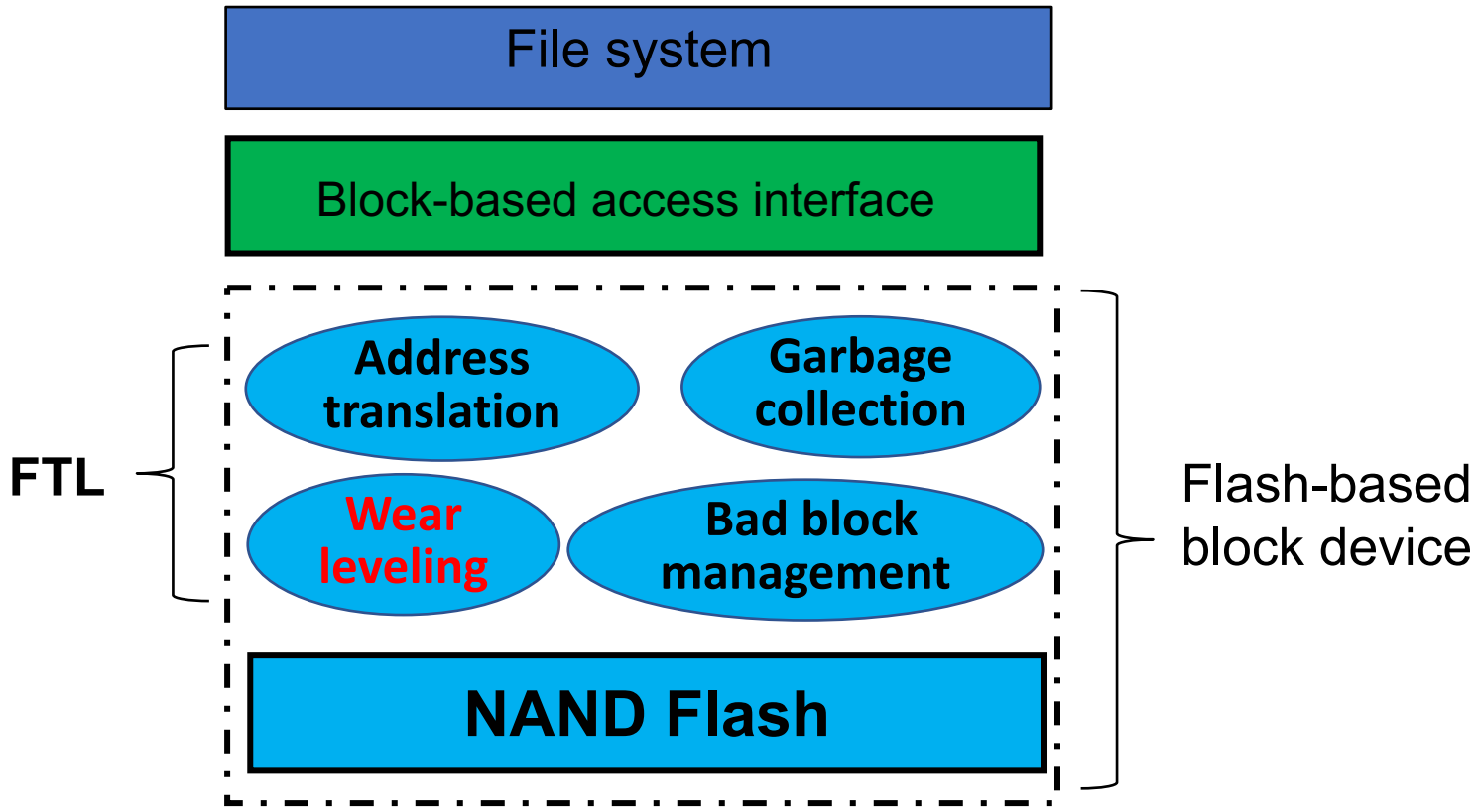
- ✓Flash memory is update unfriendly
- ✓Not prefer in-place update, but prefer out-of-place update
- ✓The blocks storing obsolete data should be reclaimed periodically by garbage collection



# Flash Translation Layer (cont.)

Wear leveling:

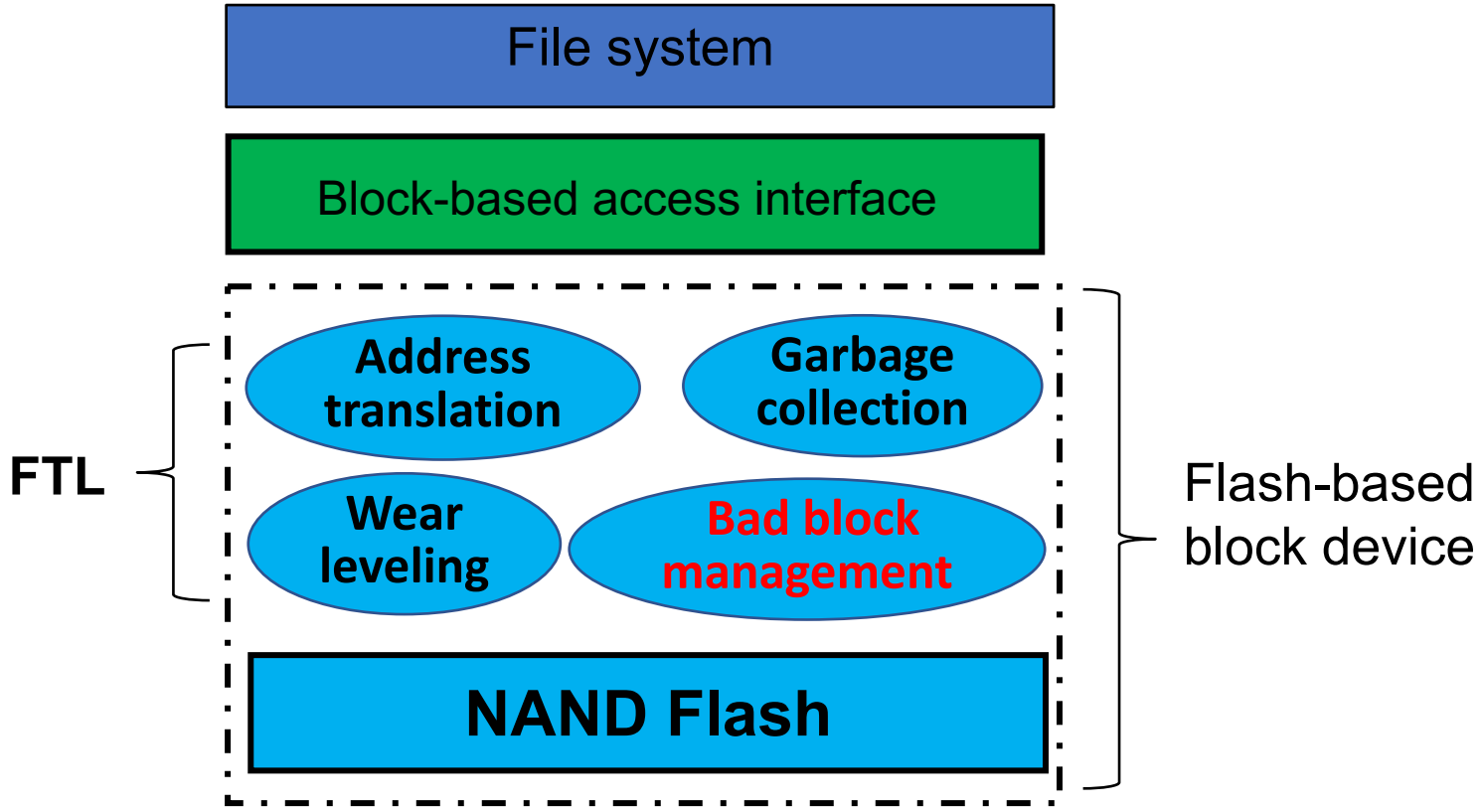
- ✓ Each flash block can be programmed/erased for a limited number of times
- ✓ Distribute writes evenly across the flash to prolong its lifetime



# Flash Translation Layer (cont.)

Bad block management:

- ✓Regardless how good is the wear leveling, some flash blocks will eventually turn “bad” and cannot reliably store data
- ✓Bad block management is to manage these bad blocks



# New Design Goals of PDE Systems for Mobile Devices

- Defend against a multiple-snapshot adversary

- Preventing deniability compromise in the flash memory by hacking into FTL firmware

# Paper Presentation

- DEFTL: Implementing Plausibly Deniable Encryption in Flash Translation Layer

- Presented by Chirag Dave