

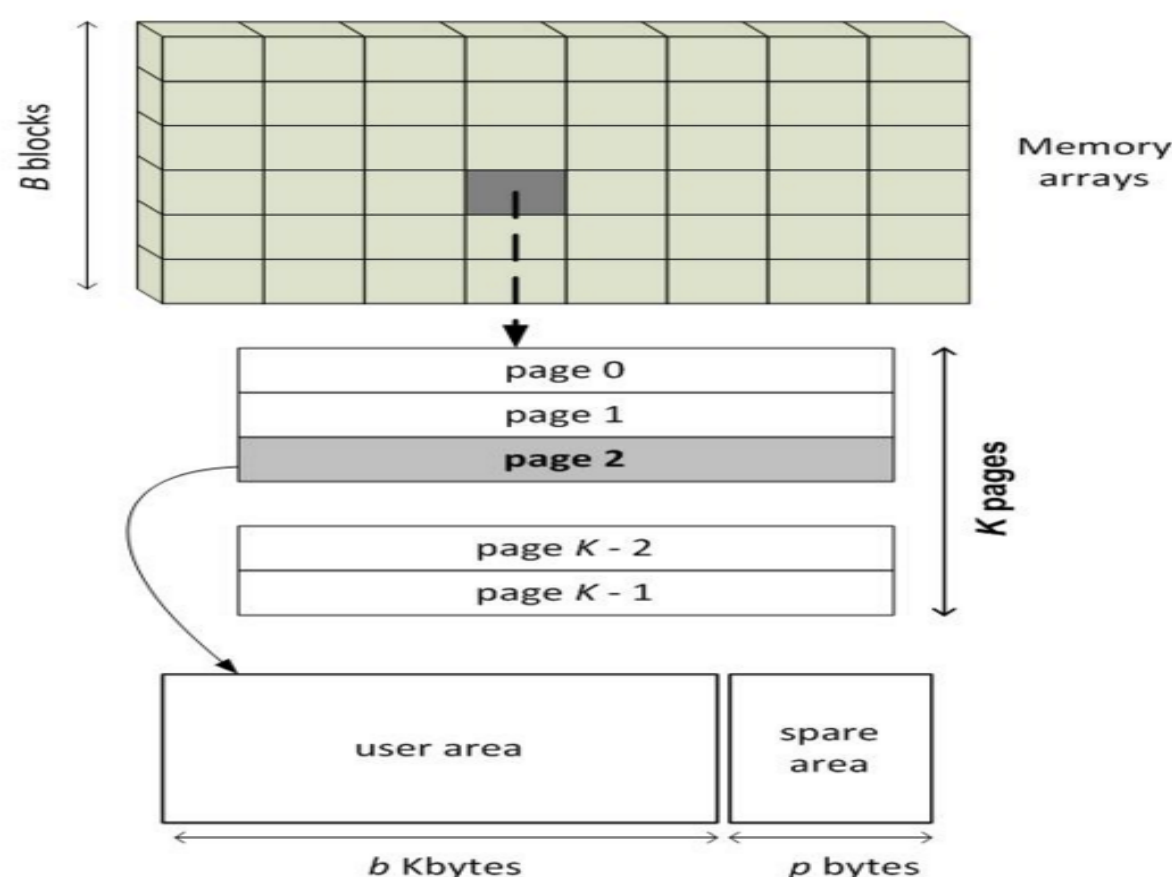
공간 활용률 향상을 위한 flash memory 최적화

충남대학교 컴퓨터 공학과

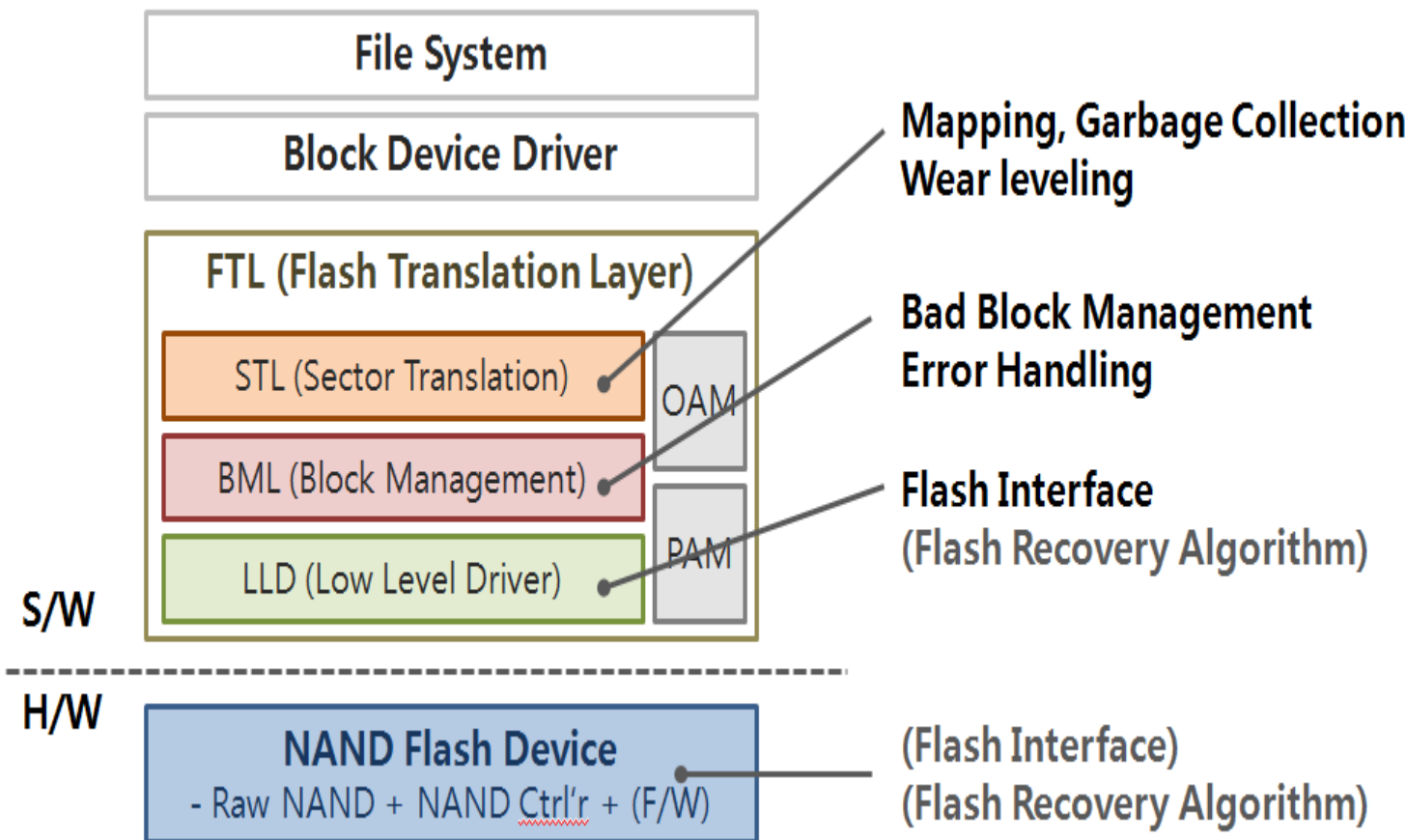
지도 교수 : 조 은 선

201002449 조원규, 201002452 위재민, 201002514 최형탁

Flash memory



Flash-based storage architecture



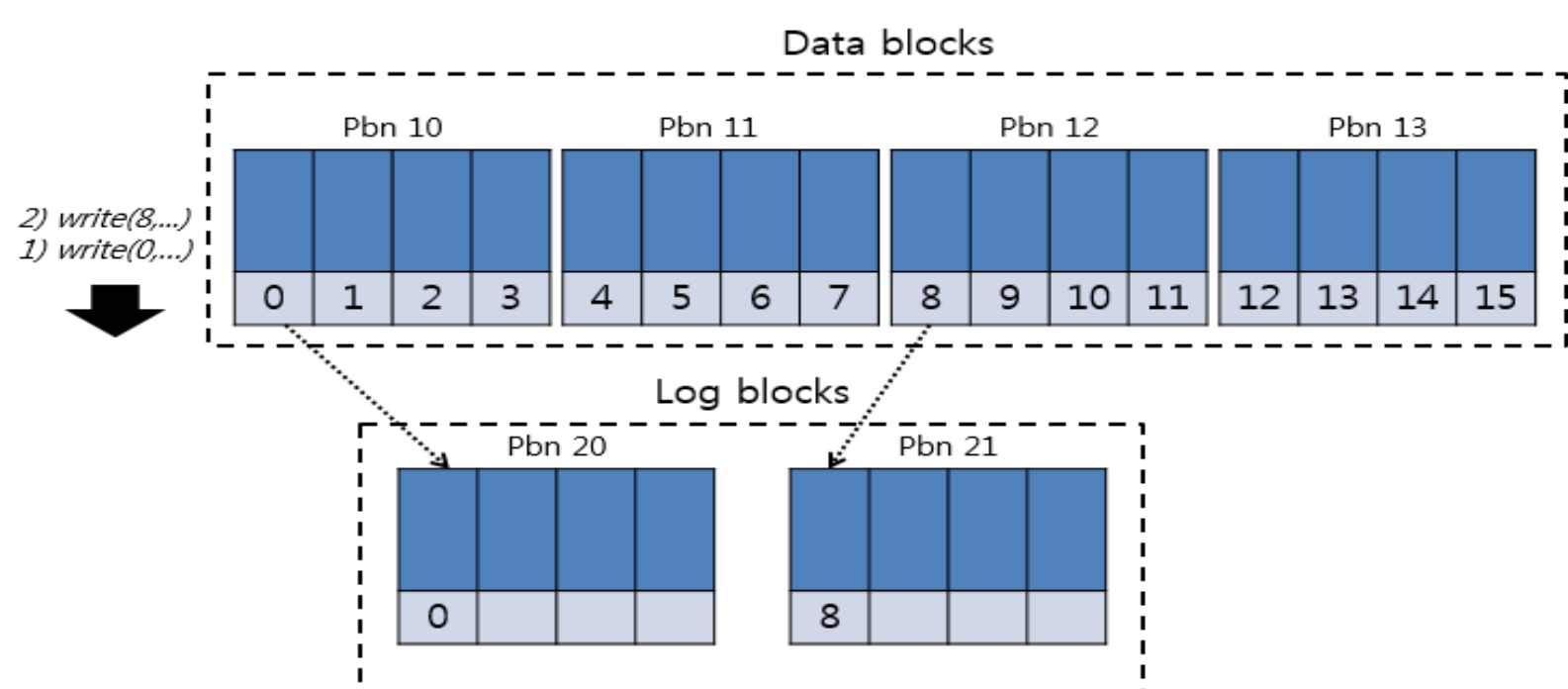
Motivation

연구 동기

- ◆ BAST : 1:1로 대응되는 구조에서 Log block 개수가 제한되어 있으므로 Log block이 필요한 Data block이 발생할 때마다 **사용 가능한 공간이 있음에도 병합 현상이 발생**
- ◆ FAST : SW log block을 관리하면서 offset이 순차적으로 들어오지 않을 때마다 **사용 가능한 공간이 있음에도 병합 현상이 발생**

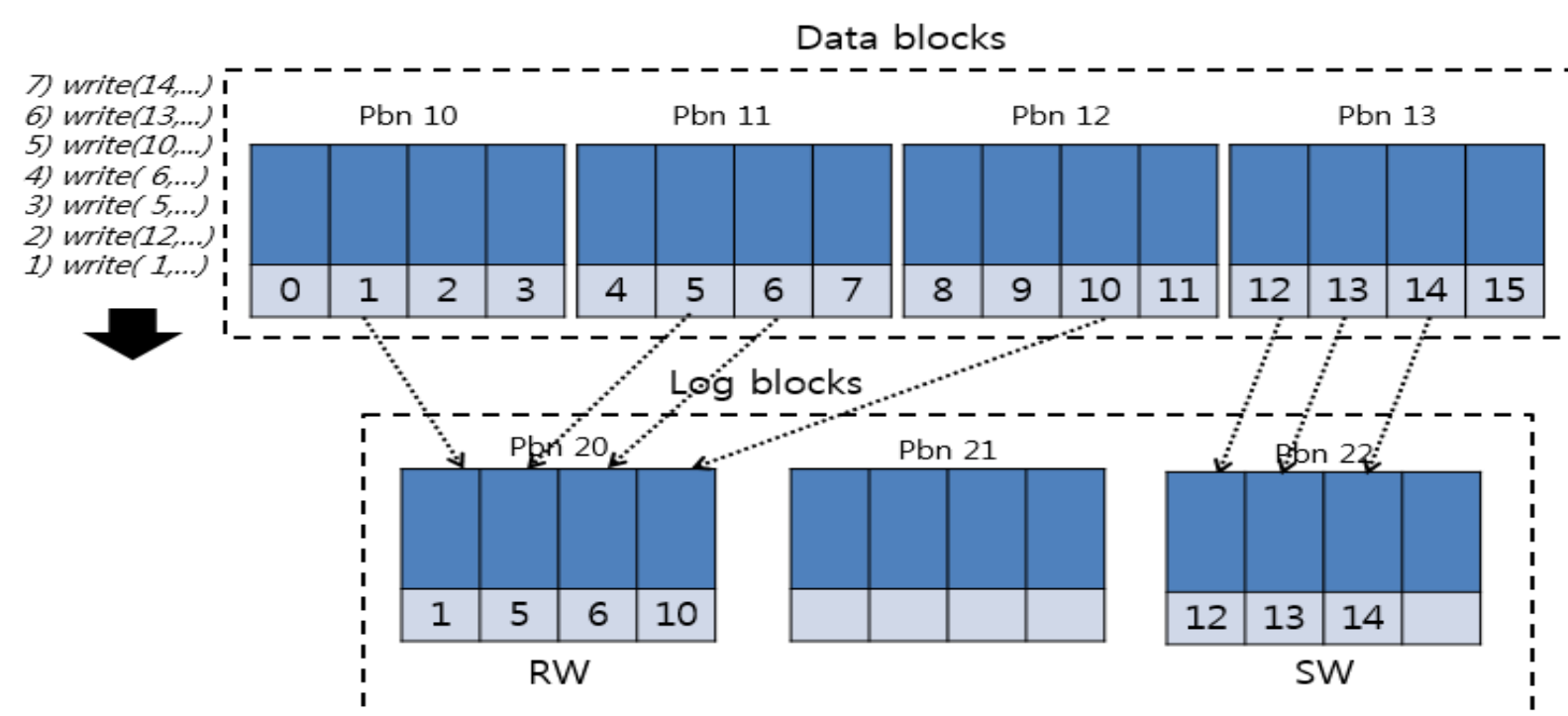
대표적인 FTL 기법인 BAST와 FAST는 데이터 블록과 로그 블록의 병합 연산 시, 로그 블록의 공간 활용률 저하 문제 발생!!!!

기존 연구 : BAST



- ◆ 임시 블록인 로그 블록과 데이터 블록이 1:1 대응관계
- ◆ 사용 가능한 로그 로그 블록이 없다면, 데이터블록과 로그블록 병합
-> 블록 내 사용 가능한 페이지가 존재함에도 병합되어 공간 활용률 저하 현상초래

기존 연구 : FAST



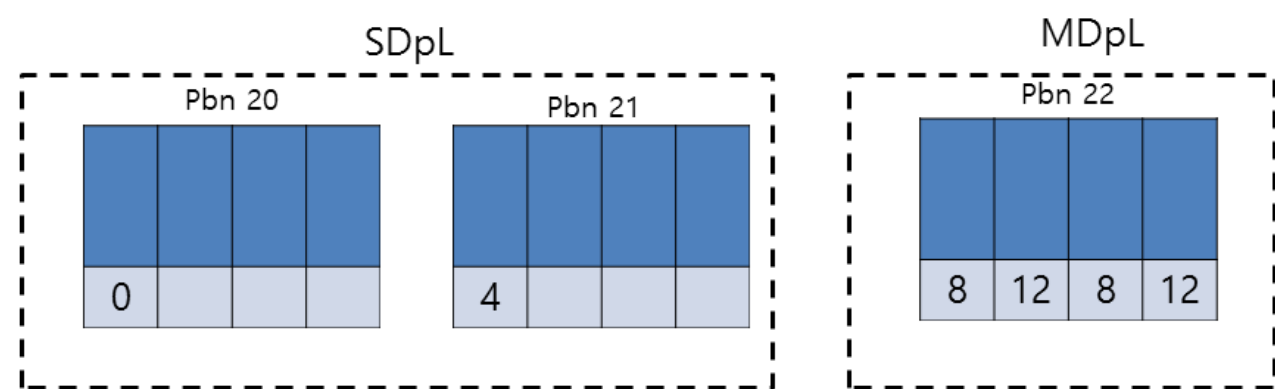
- ◆ 임시 블록인 로그 블록과 데이터 블록이 1:N 대응관계
- ◆ 랜덤 쓰기 연산 처리를 위한 RW 블록
- ◆ Offset이 0일 때 연속적인 쓰기 연산 처리를 위한 SW 블록
-> SW 블록 내 사용 가능한 페이지가 존재함에도 병합되어 공간 활용률 저하 현상초래

SMSFTL(Single data & Multiple data Swap FTL)

- ◆ 연구 목표 : 로그 블록 공간 활용률 개선을 통한 성능 및 신뢰성 향상

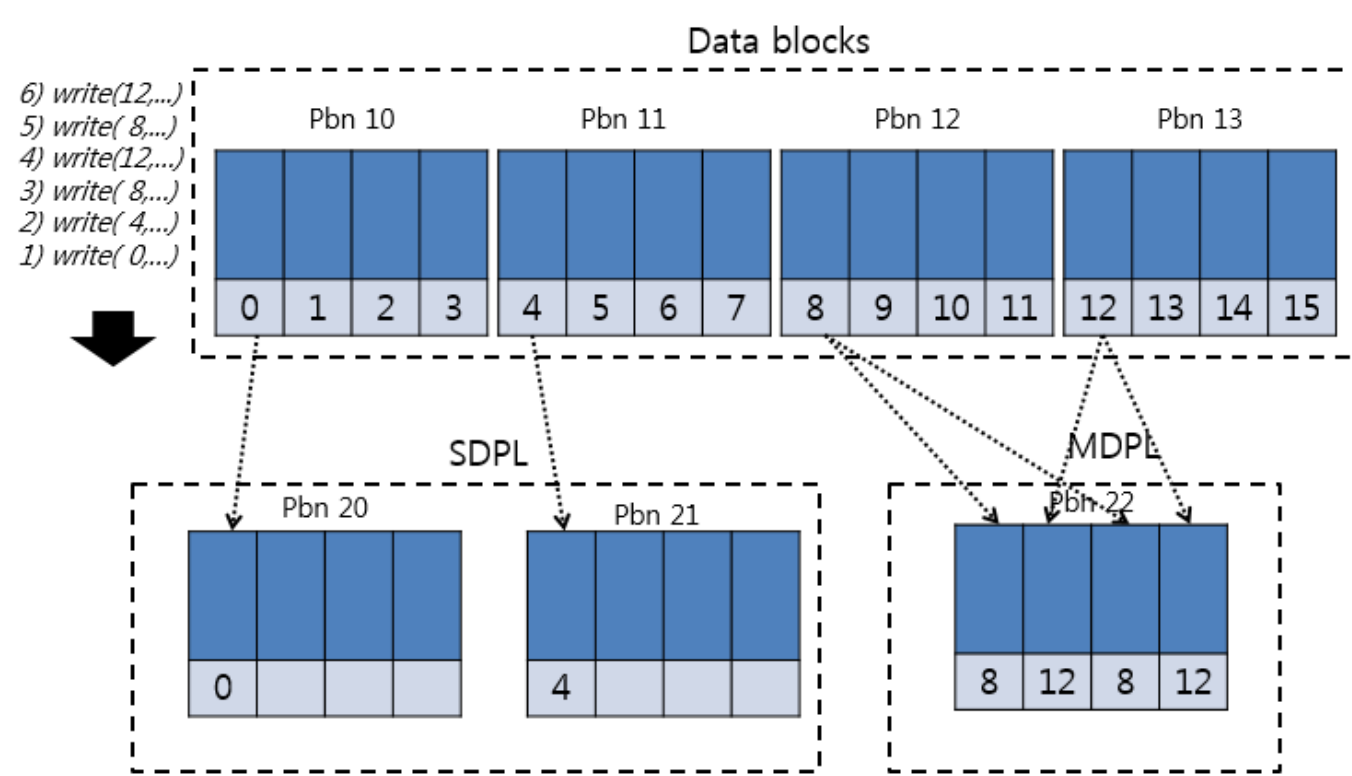
Sequential or random pattern 의 data 가 write 하였을 때 Log block 이 상황에 따라 성질이 변하는 SWAP 연산을 제안

◆SMSFTL의 로그 블록 구성

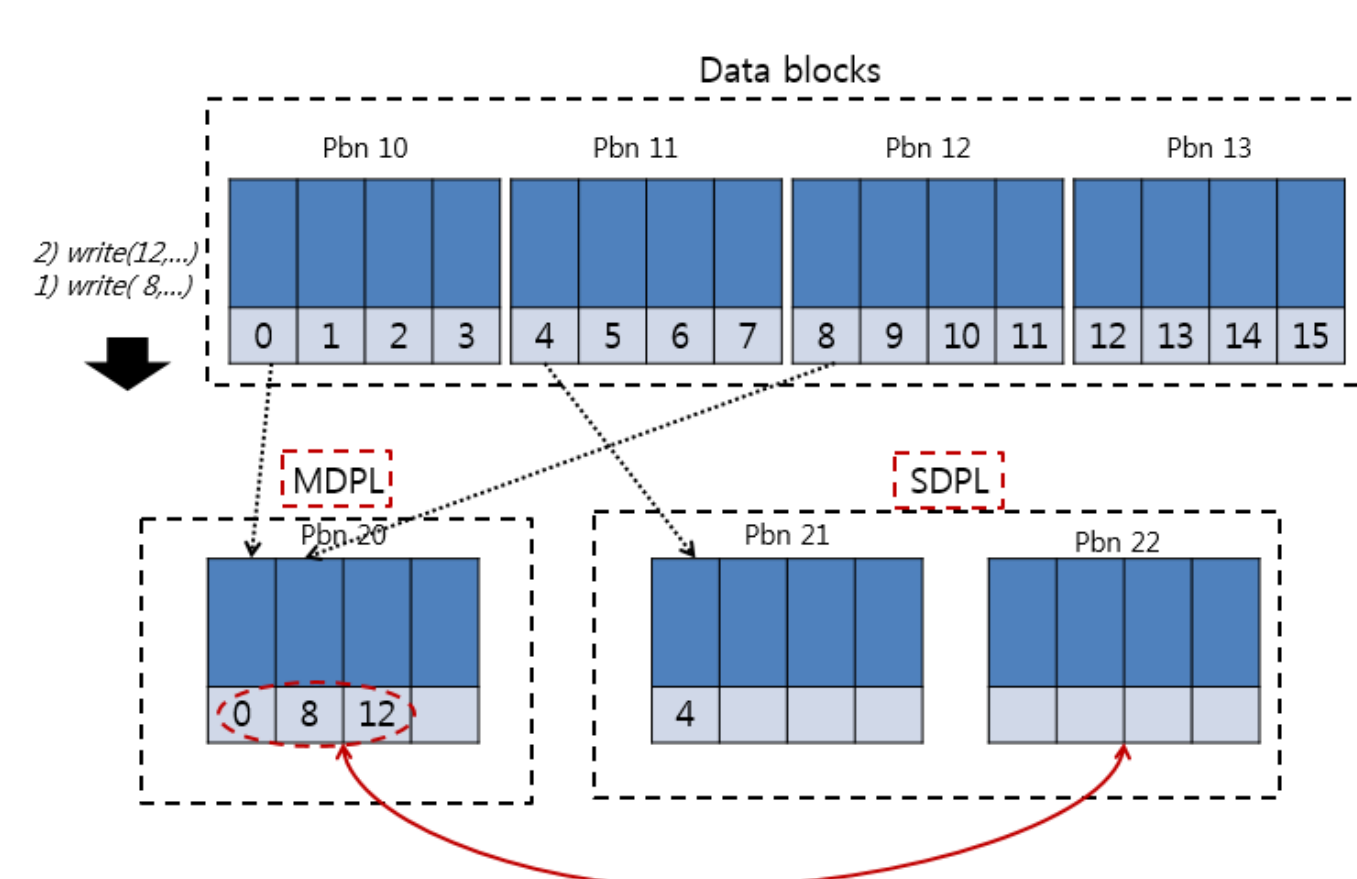


- ◆ 데이터 블록과 로그블록이 1:1로 대응되는 SDPL(Single Data Per Log Block)
- ◆ 데이터 블록과 로그블록이 1:N으로 대응되는 MDPL(Multiple Data Per Log Block)

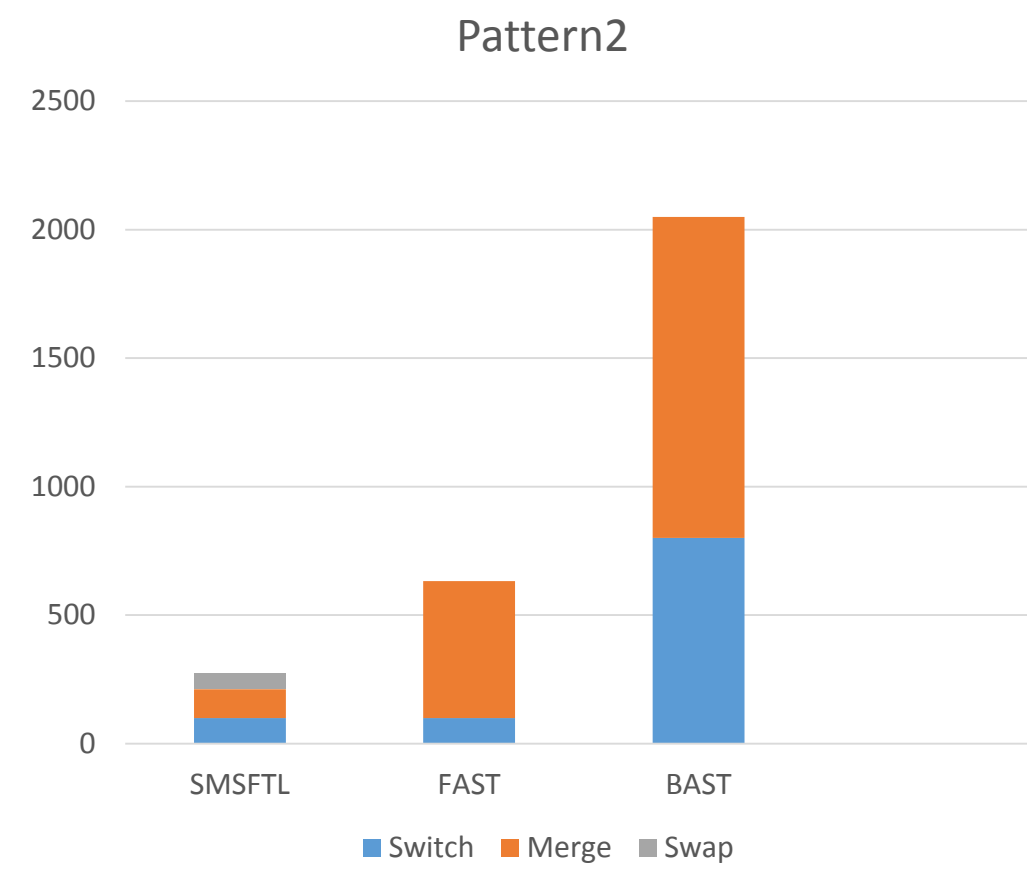
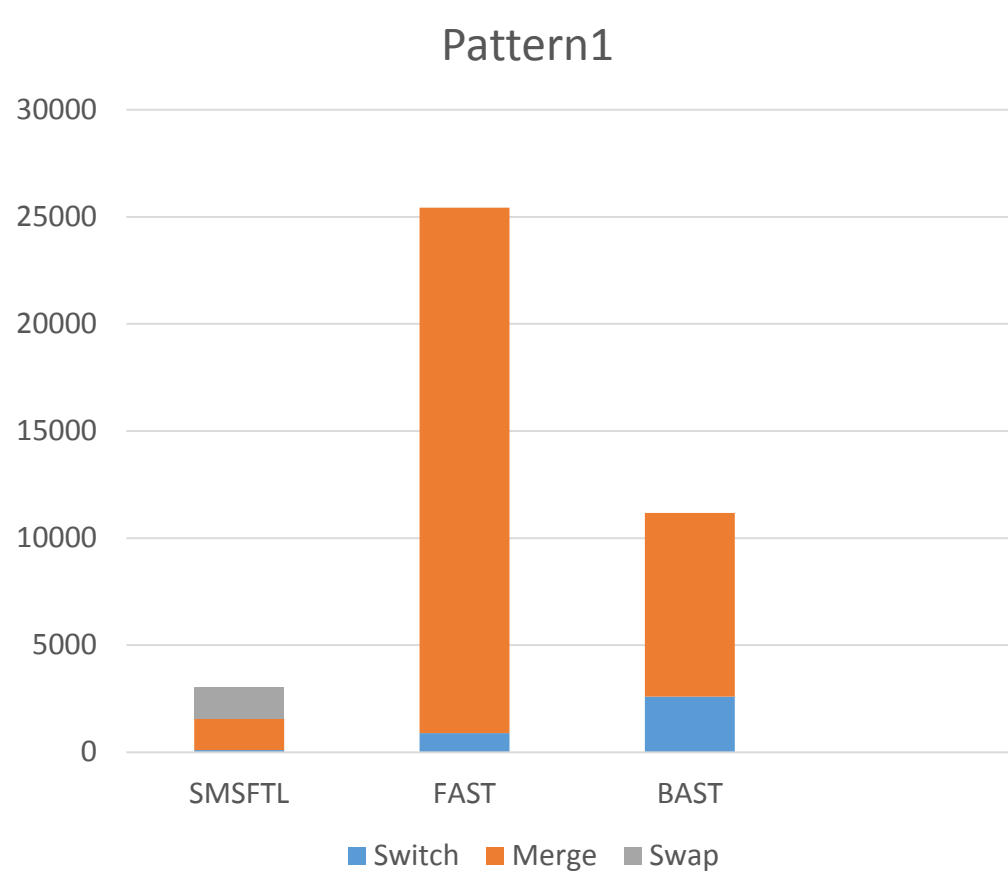
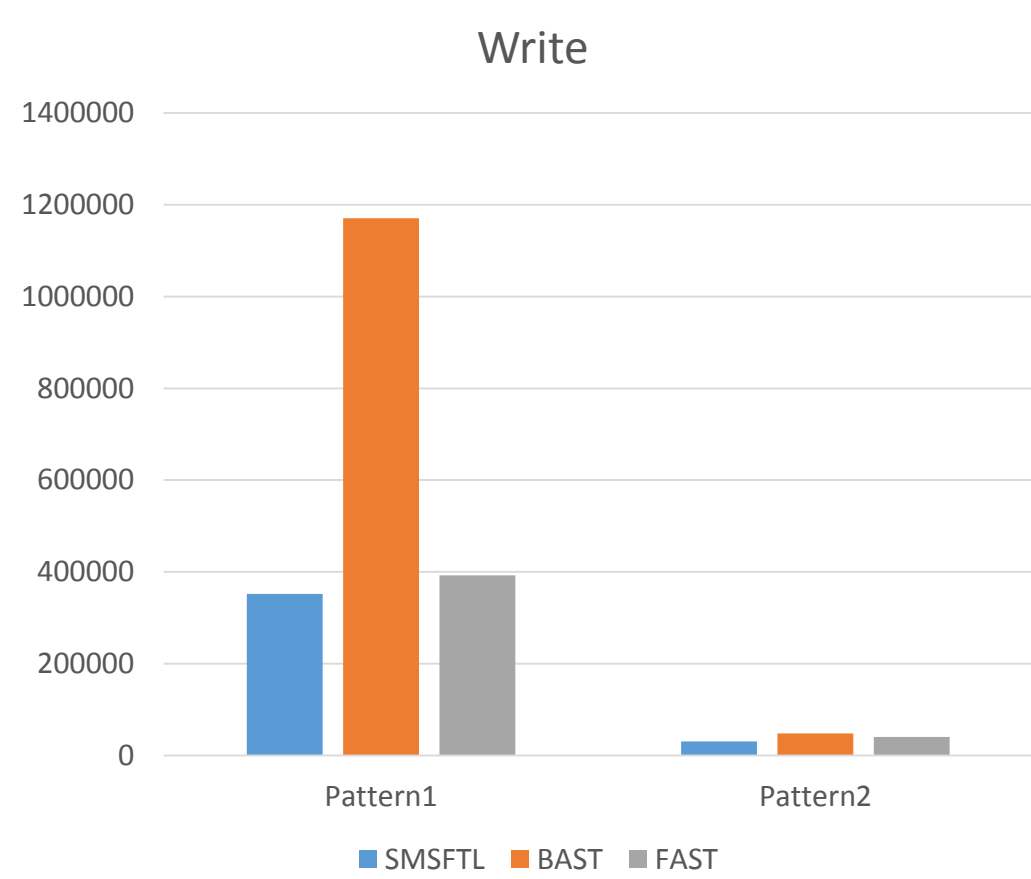
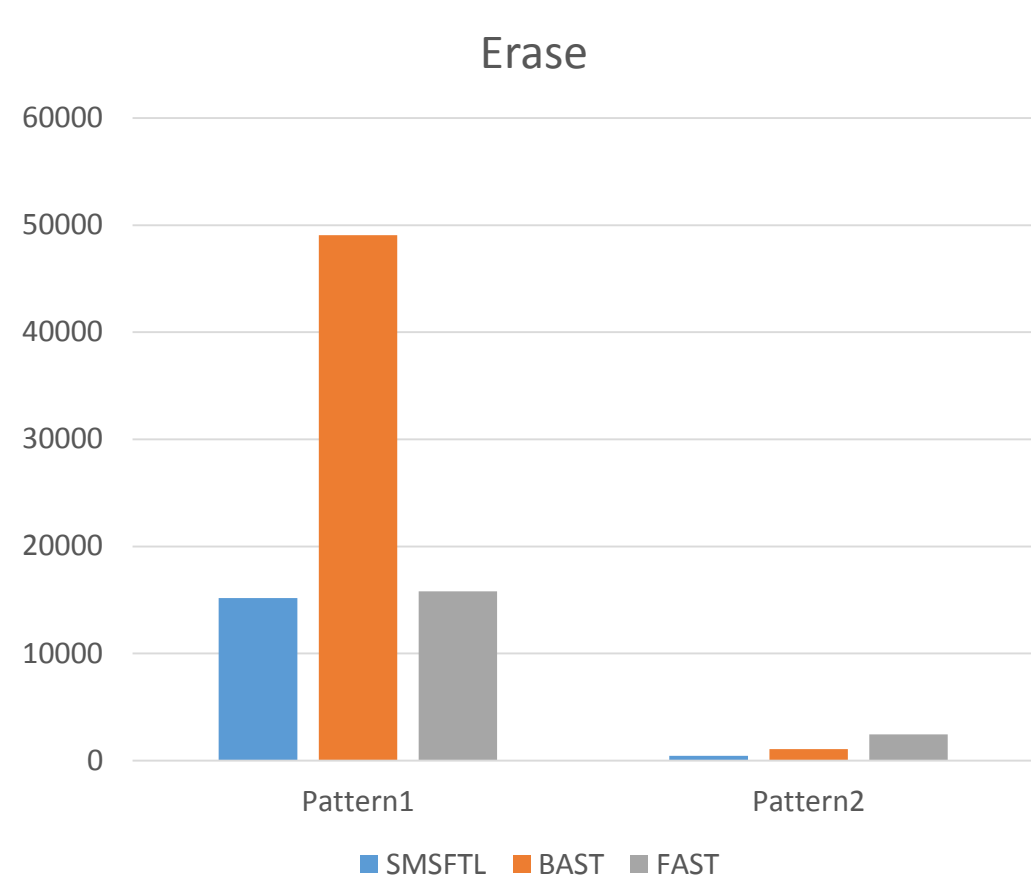
1. 사용 가능한 SDPL(Single Data Per Log block)이 있는 경우
 - 입력 write 연산을 해당되는 SDPL에 저장
 - Ex) Write 요청 0, 4
2. SDPL이 부족한 경우
 - 입력 write 연산을 MDPL(Multiple Data Per Log block)에 저장
 - Ex) Write 요청 8, 12, 8, 12



3. MDPL이 부족한 경우
 - 병합 연산 및 SWAP 동작 발생
 - **SWAP 동작:** 기본 SDPL 중에서, 하나의 로그 블록이 MDPL로 변환되어 새롭게 요청된 쓰기 연산은 변환된 SDPL에 저장됨.
 - Ex) Write 요청 8



연구 결과



Pattern 1 : random
Pattern 2 : sequential