A Delayed Synchronization Scheme for Flash Memory-Based Embedded Database Systems

Ha-Joo Song

Department of Computer Engineering

Pukyong National University

Santa Clara, CA USA
August 2007

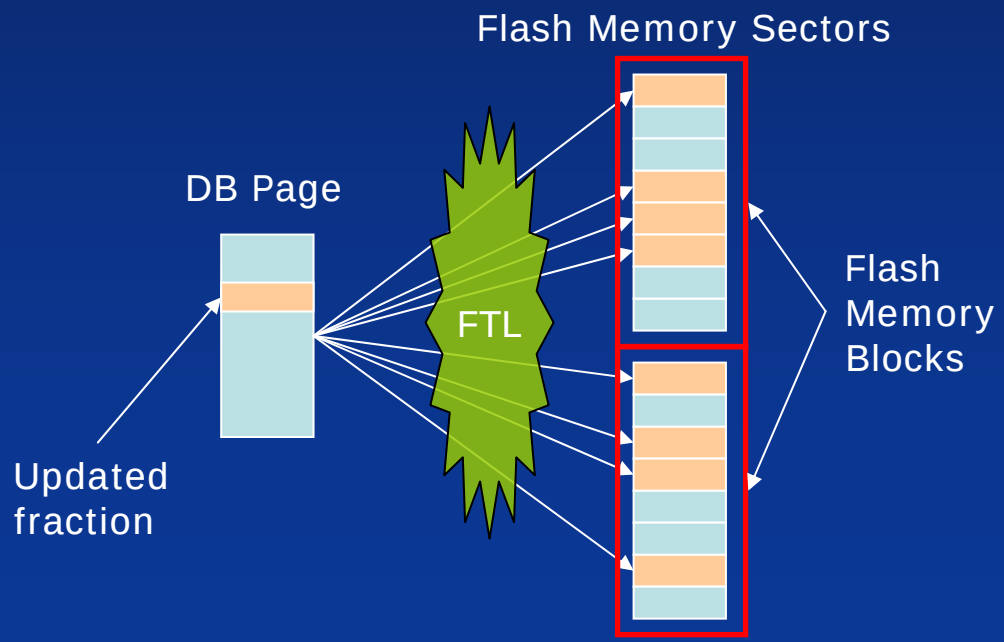
Characteristics of Flash Memory

- Asymmetric read and write access time
- No in-place update
  - Block erase operation is needed
- Little performance difference between a random access and a sequential access
- Limited number of erases on a block
  - Less updates → longer lifetime
- Observation
  - Reducing write operations is favorable both for the execution performance and the storage lifetime

Pages in Embedded Database Systems

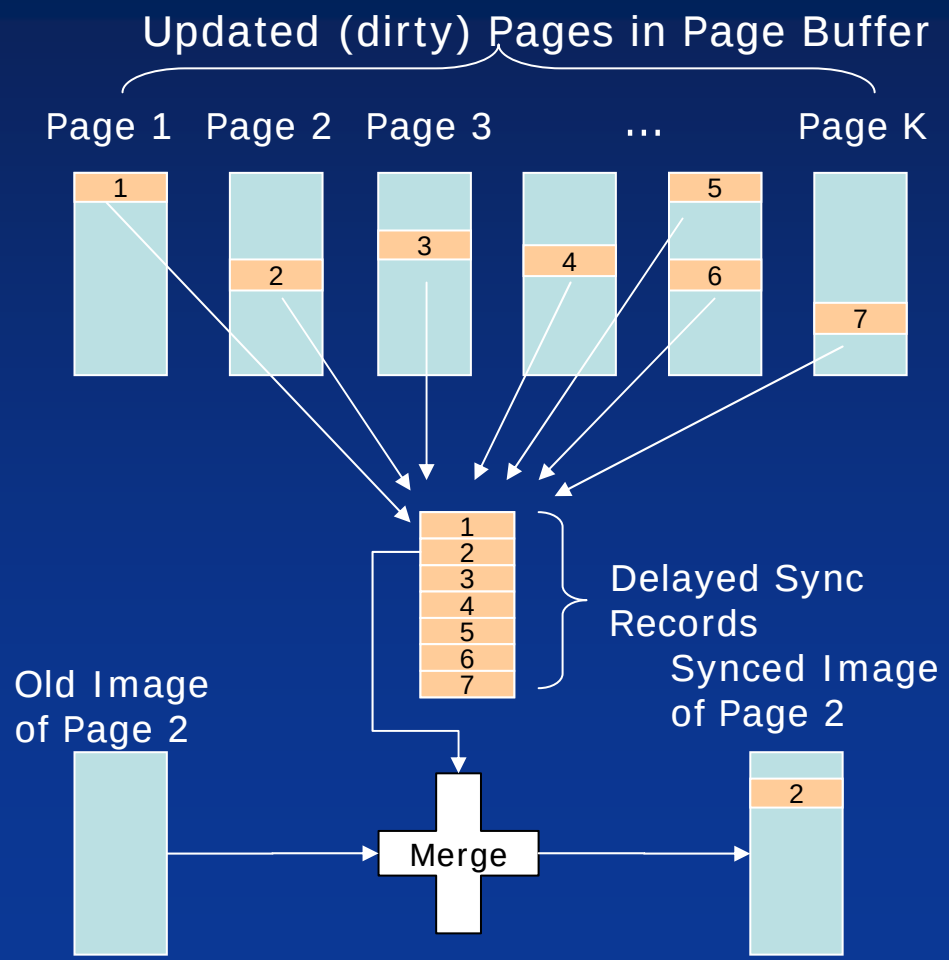
- Physical building block of the databases
- A unit of input and output to the flash memory
- Larger than a flash memory sector
  - Database page
    - 1KB / 4KB / 8KB
  - Flash memory sector
    - 512B / 2KB

- Observation
  - A buffer page is synchronized to a storage page even though very small part of it is updated

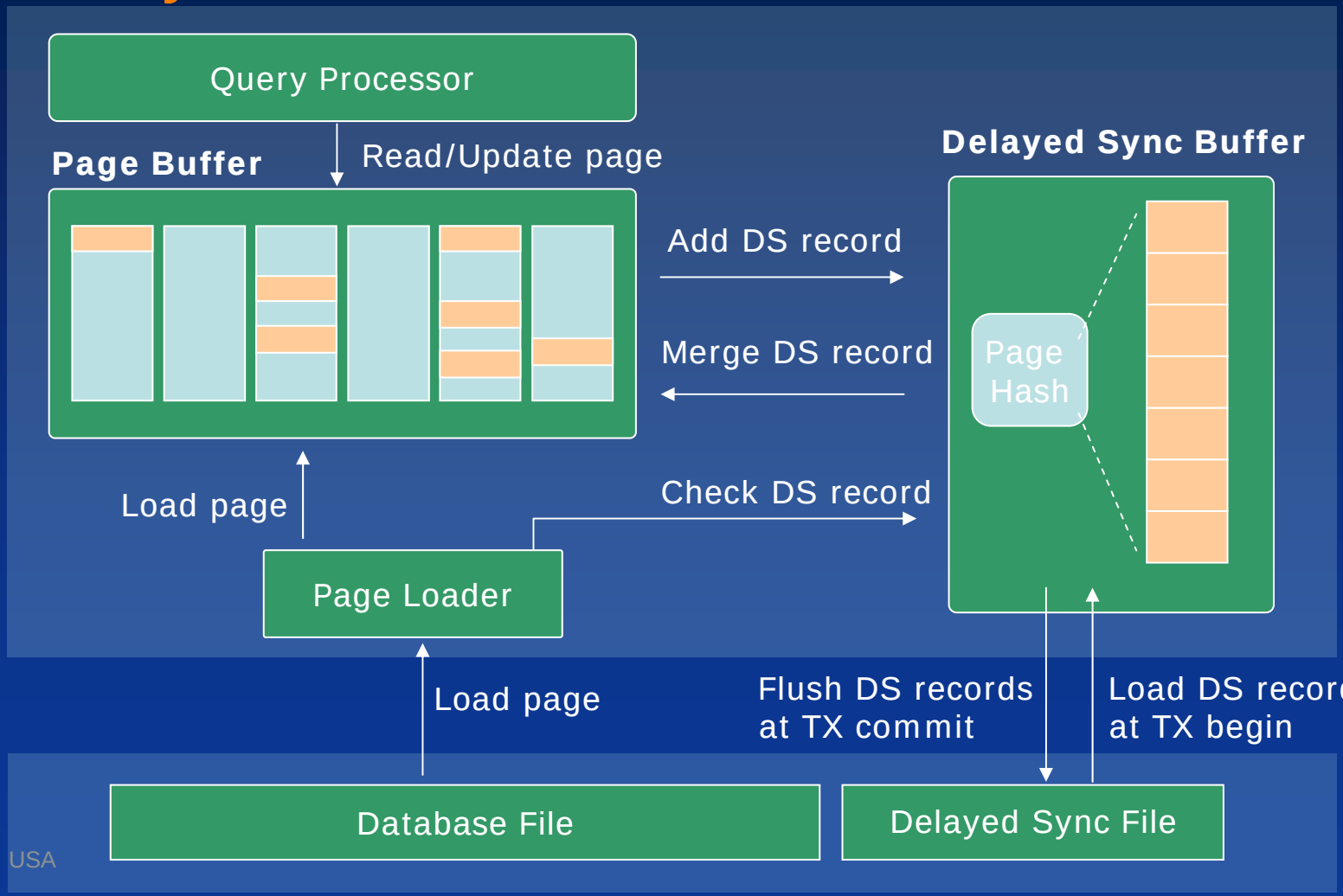


Delaying Synchronization between Buffer Pages and Storage Pages

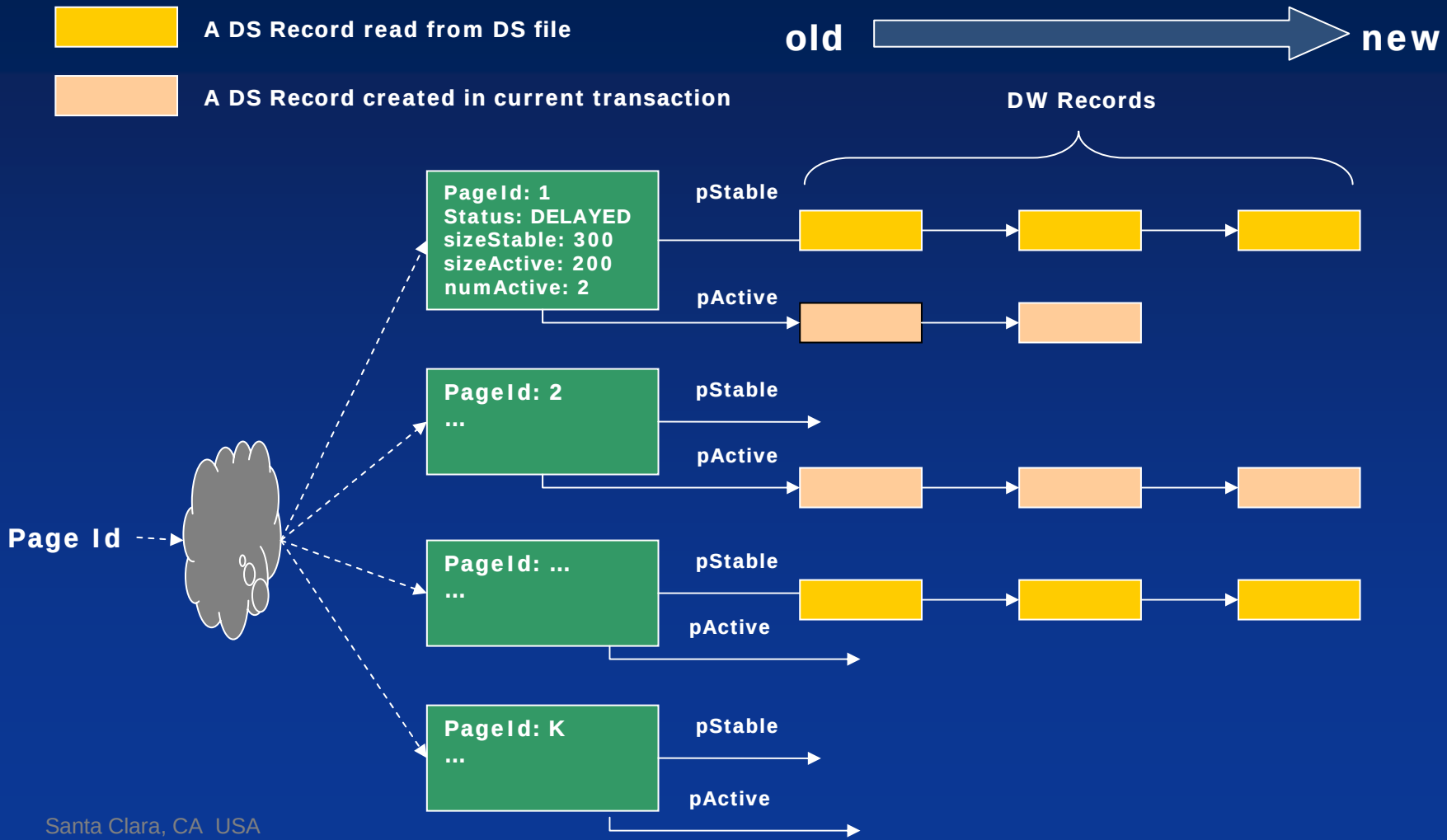
- Coalesces modified fractions of dirty pages into a single page
- Reduces flash memory sector writes
- Needs DS record merge on later page read



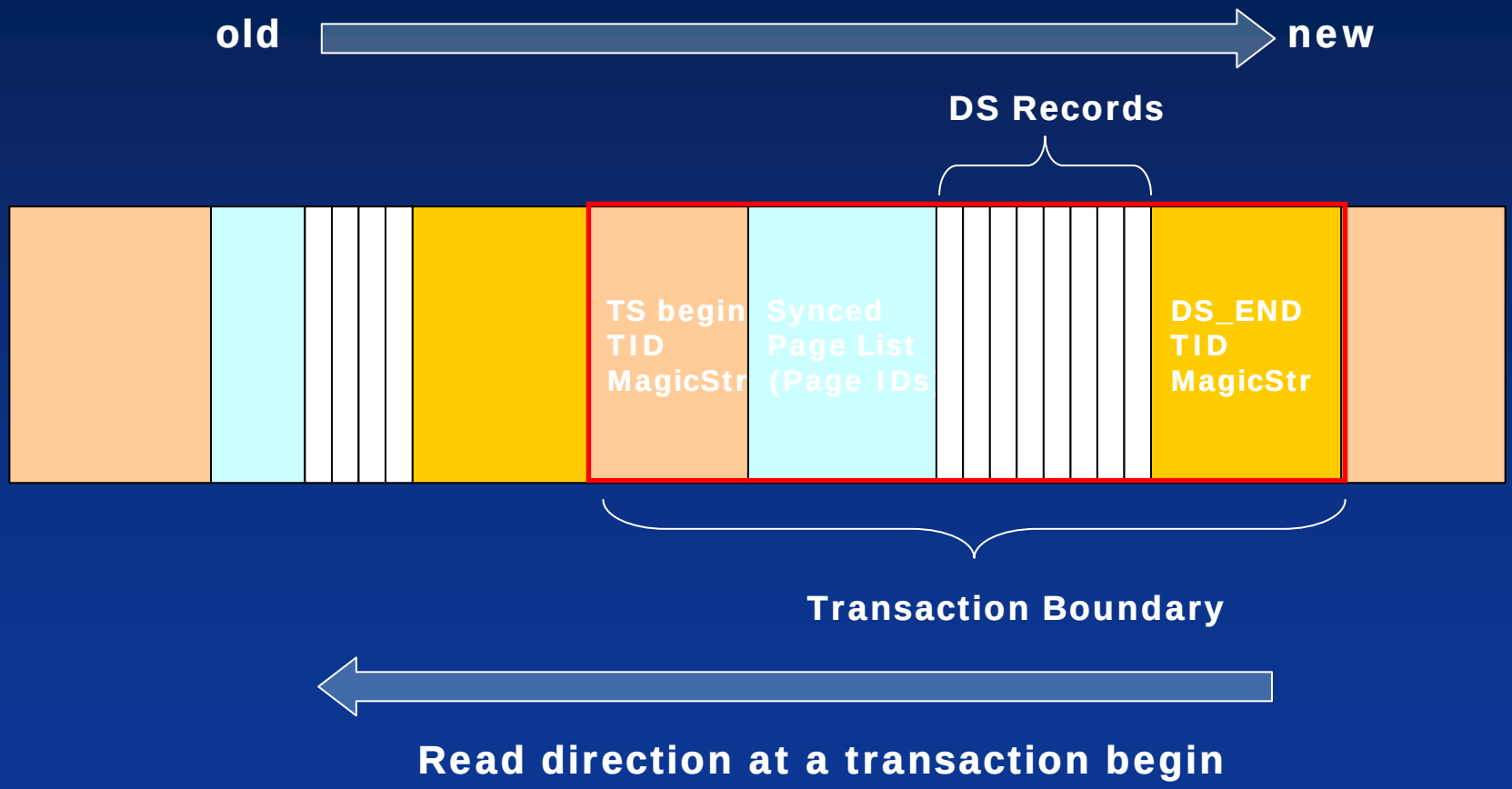
EDBMS Architecture for Delayed Synchronization



Structure of Delayed Sync Buffer



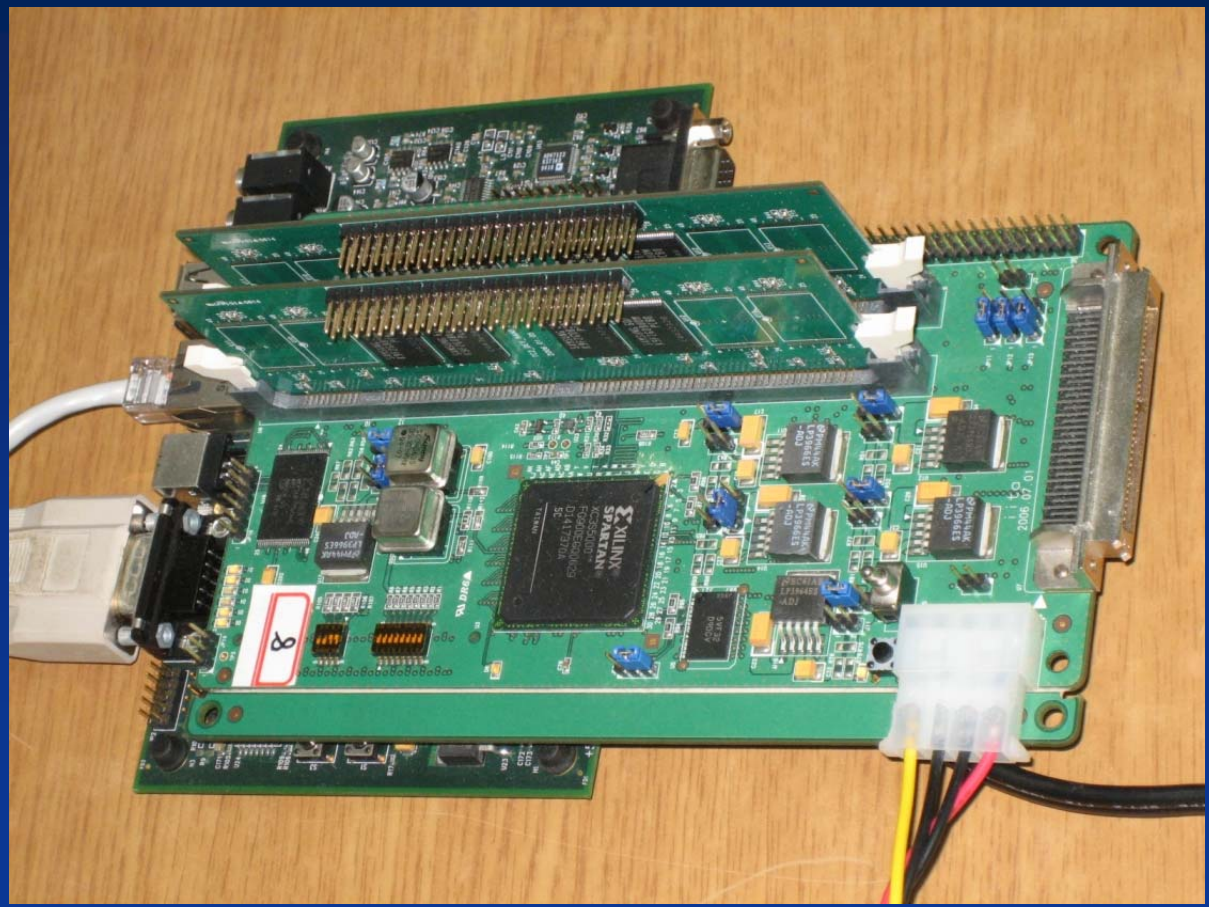
Structure of A Delayed Sync File



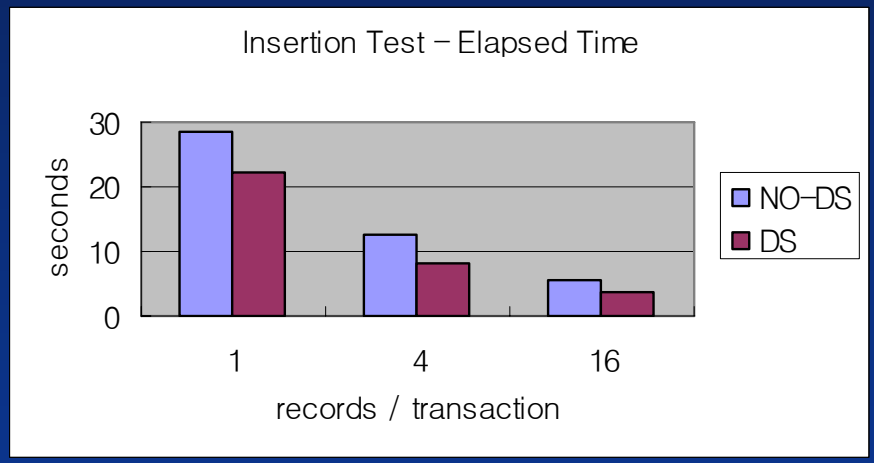
Performance Test

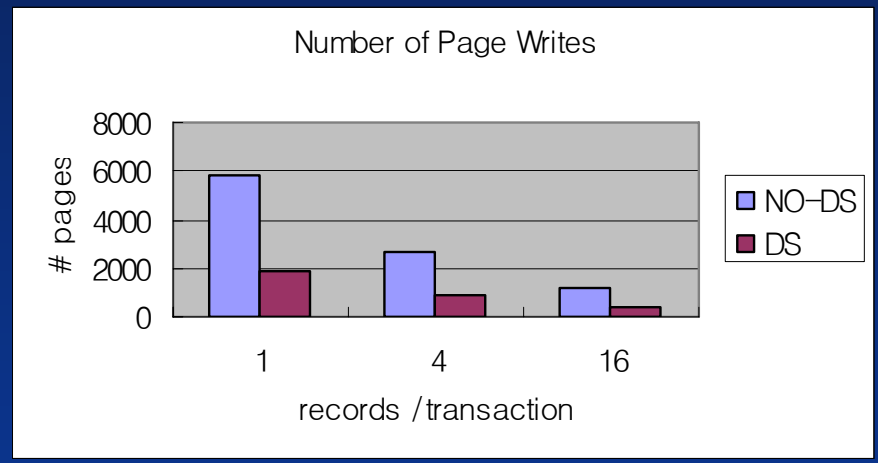
- Test Platform
  - EDBMS
    - SQLite 3.3.10
  - HW
    - Cirrus Logic EDB9315A
    - EP9315 200 MHz CPU
    - 64MB RAM
  - Flash memory
    - 1GB Flash USB
    - 128MB NAND Flash
- Test configuration
  - File systems
    - Ext2 / FAT32 / YAFFS2
  - Database
    - 4KB Page
  - 1 Table
  - Record size
    - 16B / 64B / 256B
  - Transactions
    - Insert
    - Update
    - Scan

EDB9315A with Flash Memory Daughter Board



Test Result – Record Insertion





Summary

- Characteristics of flash memory are very different from that of a disk
- Existing database techniques have to be reexamined
- Delayed synchronization scheme
  - Reduces database page writes
  - Increases write performance and lifetime of flash media
- Further research issues
  - Tightly binding delayed synchronization with FTL
  - Flash-aware recovery
  - Flash-aware temporary data processing