CS 561 Final Project

Bufferpool Implementation

By

Jiayu Lu & Joel Franklin

What is a Bufferpool?

Bufferpool is a memory area used to store a subset of the database data which is frequently accessed.

Why do we need Bufferpool?

- Reduces time to access the data. Memory access quicker than Disk access.
- Reduces number of write operations to disk
- Pinning

We understood importance of Bufferpool. How to measure performance of Bufferpool?

Bufferpool Metrics & Terms

- Page Hits/Misses
- Dirty/Clean Pages
- Read/Write IOs
- Cold/Hot Pages

A Good Bufferpool

- Increases Page Hits
- Decreases Page Misses, Read/Write IOs

How to design a Bufferpool?

- First, deciding the **Data Structure** for Bufferpool

- Second, implementing the **Page Replacement Algorithm** of Bufferpool

Bufferpool Data Structure

- Bufferpool - A vector of ‘**Pages**’

- Page - A user defined data structure with 5 attributes namely pageId, timestamp, dirty, cold, content

Page Replacement Algorithms Implemented

- LRU (Least Recently Used)
- CFLRU (Clean-First LRU) (2 Approaches)
- LRU-WSR (LRU Write Sequence Reordering)

LRU (Least Recently Used)

- **Principle** - Page which has not been accessed for the longest time (minimum timestamp) is least likely to be accessed

- **Time Complexity** = O(n) where n is maximum size of Bufferpool

CFLRU (Clean-First LRU)

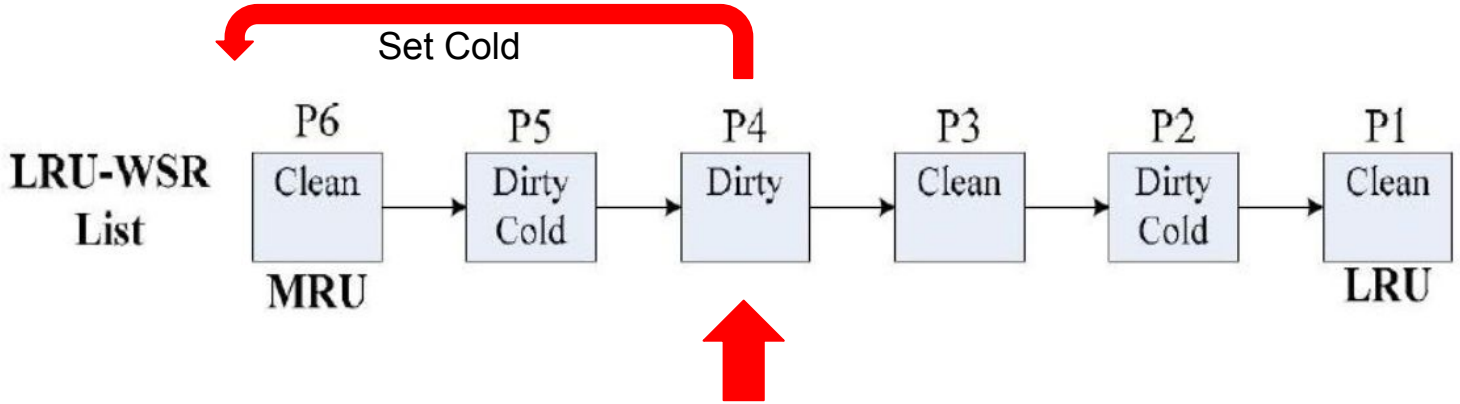
- **Principle -** Variation of LRU algorithm which takes into account the ‘dirty’ or ‘clean’ status of page

- Approach 1 - Using an **Array**
- Time Complexity = O(n * logn) where n is maximum size of Bufferpool

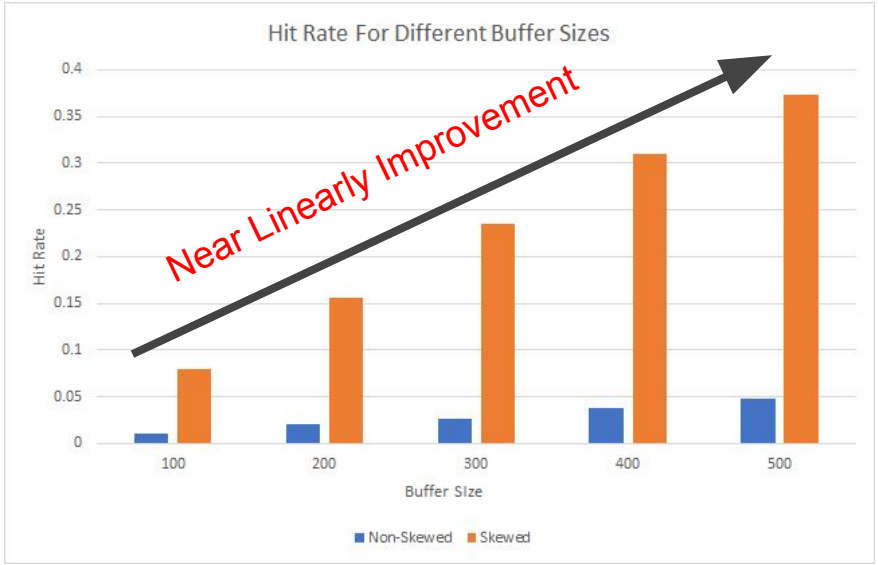
- Approach 2 - Using a **Min Heap**
- Time Complexity = O(n * logn) where n is maximum size of Bufferpool

LRU-WSR (LRU Write Sequence Reordering)

Principle - Delays the eviction of **Dirty Hot** pages



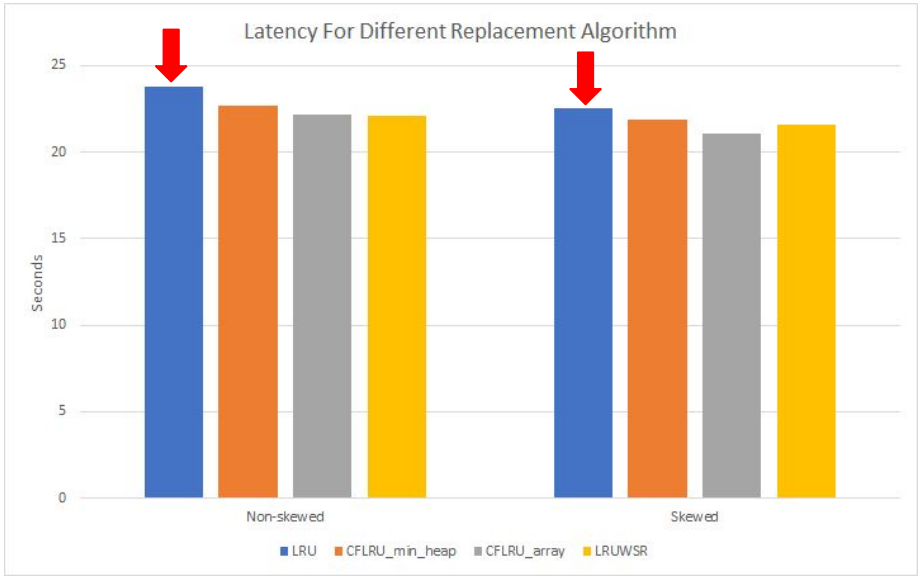
Hit Rate Evaluation on Buffer Size



70% Reads, 30% Writes
Skew data: 90% operations on 10% data

For skewed: 7.7~8.7 times higher hit%

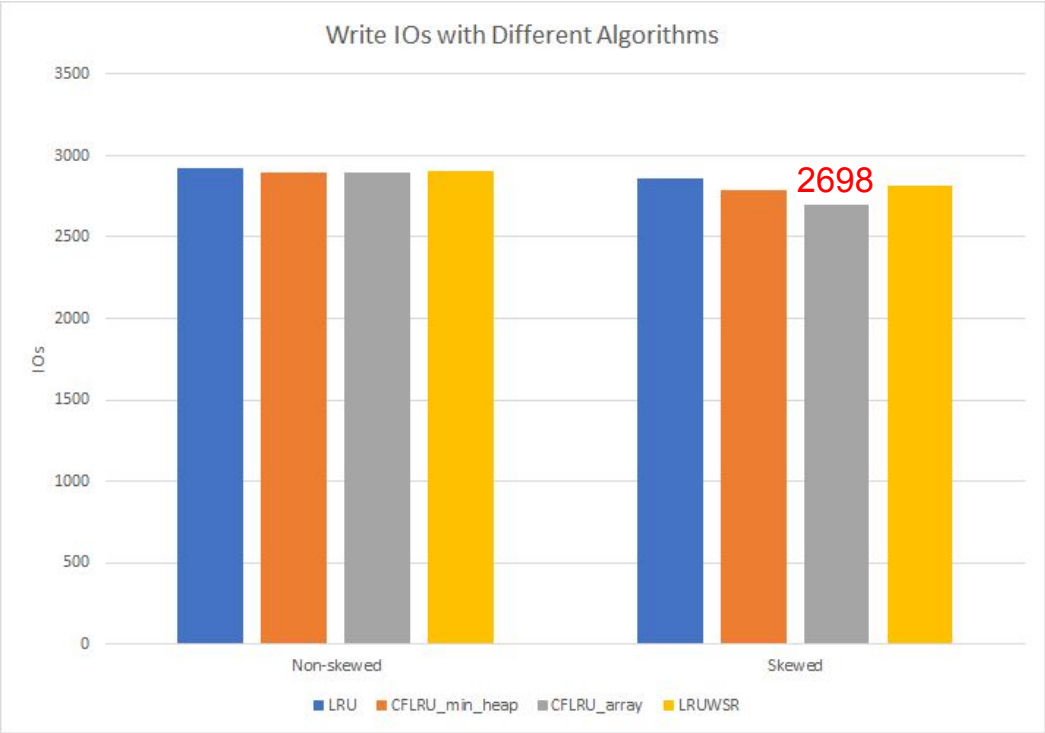
On-disk Latency Evaluation



70% Reads, 30% Writes
Skew data: 90% operations on 10% data

Default LRU has the largest latency

On-disk IO Evaluation



3000 Write Operations
->2698 Write IOs ~0.9x