ECS 165 MILESTONE 2

Group 14

Ayman Dewan, Bin Lama, Nathan Ng, Kyle Insaurralde, Raynier Tan

Bufferpool

- Keeps a limited amount of pages in memory
    - Number of memory slots set in config
- Stores and loads pages from disk
- Slot information:
    - dpage: physical location of page on disk
    - pins: number of active accesses
    - last_accessed: time slot was last accessed
    - dirty: was data changed
    - data: bytearray
- Least recently used eviction policy

Bufferpool: Access Page to Release Page

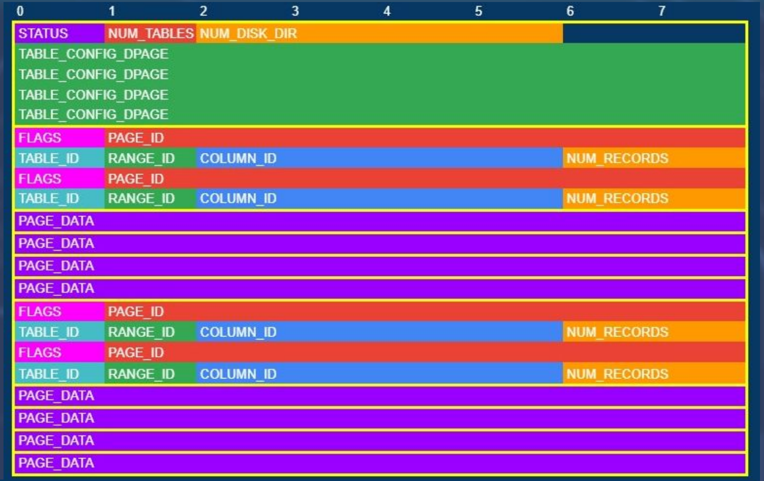
1. Page requests its data from Bufferpool
2. Load page into memory if not already
   a. First, check if any slots are free
   b. If not, find least recently accessed slot that is not in use
   c. Evict previous slot data and write to disk
   d. Load current pages data from disk into slot
3. Mark slot as dirty if write and set last_accessed time
4. Increase pin count on slot
5. Return reference to data in memory
6. Page completes its read or write
7. Page tells Bufferpool it can be released

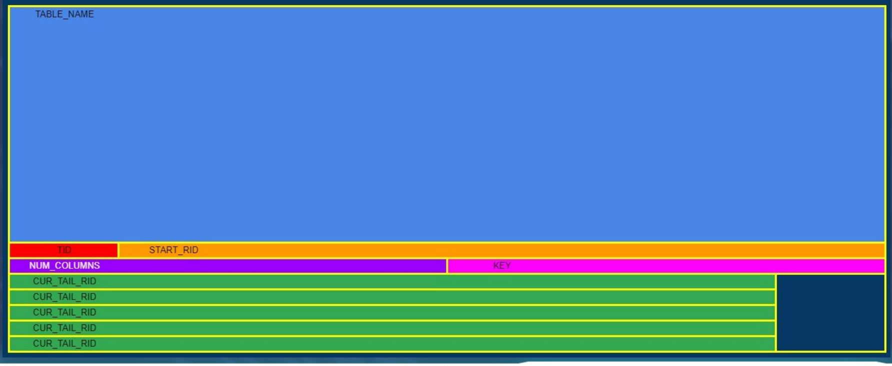
Disk

	dpage	Byte #	0	1	2	3	4	5	6	7
DB_CONFIG	0	0	STATUS	NUM_TABLES	NUM_DISK_DIR					
TABLE 0	0	8	TABLE_CONFIG_DPAGE							
TABLE 1	0	16	TABLE_CONFIG_DPAGE							
...	0	...	TABLE_CONFIG_DPAGE							
TABLE 254	0	2048	TABLE_CONFIG_DPAGE							
DIRECTORY 1 ENTRY 1	1	4096	FLAGS	PAGE_ID						
	1	4104	TABLE_ID	RANGE_ID	COLUMN_ID				NUM_RECORDS	
DIRECTORY 1 ENTRY 256	1	8176	FLAGS	PAGE_ID						
	1	8184	TABLE_ID	RANGE_ID	COLUMN_ID				NUM_RECORDS	
Page from entry 1	2	8192	PAGE_DATA							
Page from entry 2	3	12288	PAGE_DATA							
...	...	...	PAGE_DATA							
Page from entry 256	257	1052672	PAGE_DATA							
DIRECTORY 2 ENTRY 1	258	1056768	FLAGS	PAGE_ID						
	258	1056776	TABLE_ID	RANGE_ID	COLUMN_ID				NUM_RECORDS	
DIRECTORY 2 ENTRY 256	258	1060848	FLAGS	PAGE_ID						
	258	1060856	TABLE_ID	RANGE_ID	COLUMN_ID				NUM_RECORDS	
Page from entry 1	259	1060864	PAGE_DATA							
Page from entry 2	260	1064960	PAGE_DATA							
...	...	...	PAGE_DATA							
Page from entry 256	514	2105344	PAGE_DATA							

Restore Database

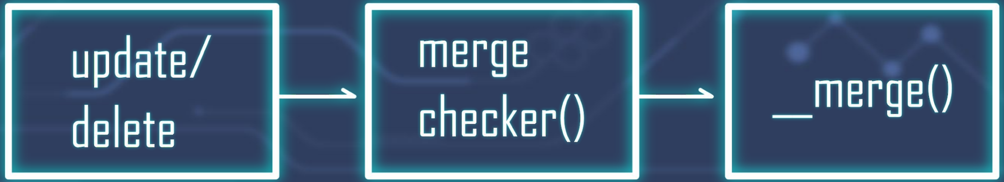
1. Read number of tables and number of directories
2. Read through disk directories
    a. Save table_id, range_id, page_id and column num of each entry
3. Restore tables
    a. Read table_name, cur_rid, num_columns, and key column





Merging

- Page range level
- Base_RID and Latest TS Column added
- Uses stack for all tail RID’s updates (FIFO)
- Merge_checker checks if merge needs to be initiated
  - It will then set the pre_tps to the rid of the latest update in the stack.
- The merge begins in a separate thread and runs in the background of the main thread.



__merge( )

- Acquires a lock, and from the unmerged_stack, finds latest Tail RID and corresponding Base RID, and adds it to a hash table
- Then, iterate over base pages and make deep copies of the base pages (from disk) to our *consolidated_page_list*
- Next, iterate over the *latest_updates* and write tail's column to consolidated page if schema for column is 1
- The base records schema is updated to show which columns have been merged into the base page.
- Lastly, append the *consolidated_page_list* to the *merge_ready_list*

Merging Consolidated Pages

- Upon any query or when a merge is ready to initiate, check the *merge_ready_list* it if has any consolidated pages ready
- If so, call *merge_page_directory()* to begin merging the consolidated pages into the page range
- Then, save the bufferpool slots if anything is dirty, and pop the *merge_ready_list* to get the consolidated pages
- Set the bufferpool for each of the consolidated pages and insert its data into the disk
- Lastly, make the page range point to the new consolidated pages and set the page range's TPS to what Pre_TPS was.

Update, Select, Sum, & Delete

- Update: For each record, it’s first tail record after merge is initiated will reset schema to 0 and reset cumulation
- Delete will create “copy” records for any non-updated columns, and then create a deleted record with schema 0 and NULL column values
- Deleted records will be shown on the base page after merged
- Sum now uses select() to retrieve values from the table
- Select is now updated to read from the new locations depending on Pre_TPS, TPS, and indirection

Update - Cases

BASE

RID	INDIRECTION	SCHEMA	k	a	b	c
5	165	0001	k_{I}	a_{I}	b_{I}	c_{II}

TAIL

	RID	INDIRECTION	SCHEMA	k	a	b	c
	. . .						
	90	91	0001	Ø	Ø	Ø	c_{I}
TPS→	89	90	0001	Ø	Ø	Ø	c_{II}
	. . .						
	65	89	0100	Ø	a_{I}	Ø	Ø
Pre-TPS→	64	65	0100	Ø	a_{II}	Ø	Ø
	. . .						
overall schema 0101	59	64	0010	Ø	Ø	b_{I}	Ø
	58	59	0010	Ø	Ø	b_{II}	Ø

Select - Race Condition Example (Pre-TPS)

BASE

RID	INDIRECTION	SCHEMA	k	a	b	c
5	65	0000	k_1	a_1	b_1	c_1

TAIL

RID	INDIRECTION	SCHEMA	k	a	b	c
...						
TPS = 100						
...						
90	5	0010	Ø	Ø	b_2	Ø
Pre-TPS = 80						
65	90	0100	Ø	a_2	Ø	Ø

Index

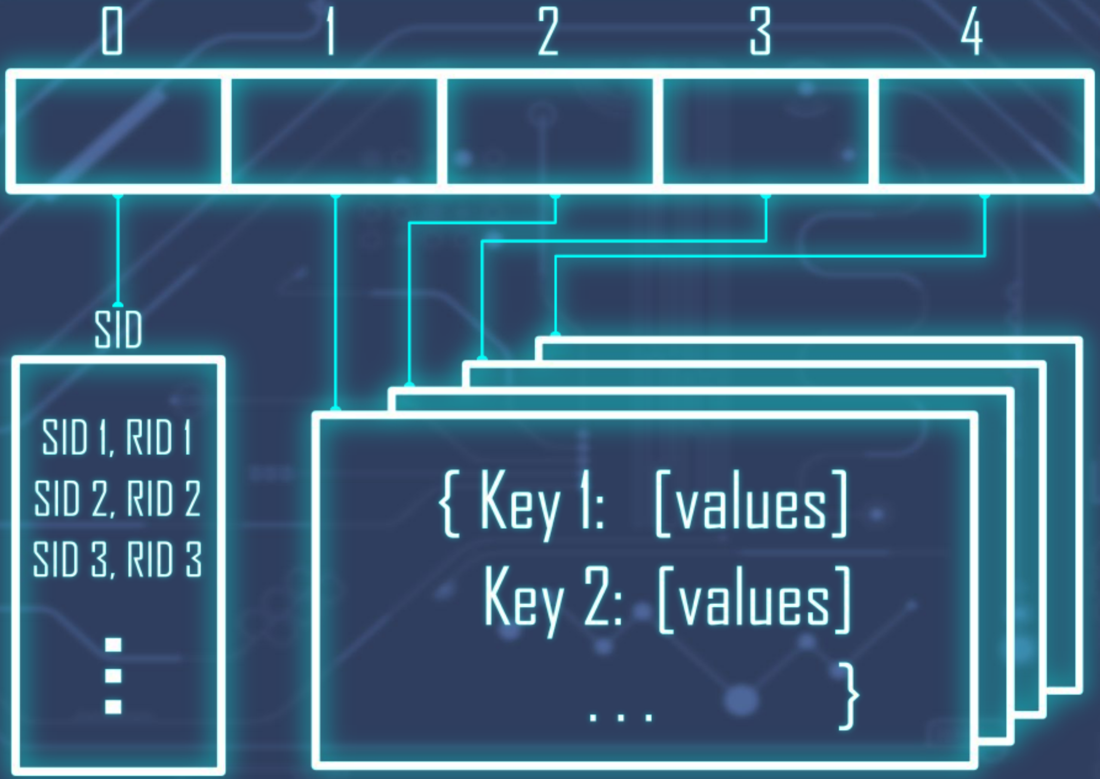
- Initially we create a list of None values. Then on the 0th index, we create a B+ tree that will have the key and value in sorted order. After that, when we insert keys in the queue, we will create an index for every keys added and assign their rid as their Value.
- If we are inserting to a different column number other than 0 (where grades will be instead of student id), we will create a dictionary by calling a create index function and in there we will add an array of values for every keys because we know that we can have different values for the same keys which is not supported on the implementation of B+ trees.

Index

- When we call the db.py to close we will close all the databases that we created that had index information saved inside them.
- In addition, when we access them next time, we will open them and if they have the keys already, then we will replace them with the new values.
- When we call drop index, we will delete the database that we have created and for the dictionary we will assign the column on which they reside to none so that they will no longer be accessed.

Index

- Only SIDs have B+ trees
- Grades contain a dictionary with a list of values


0
1
2
3
4
SID
SID 1, RID 1
SID 2, RID 2
SID 3, RID 3
{ Key 1: [values]
Key 2: [values]
... }