

# 資料庫

## 目標

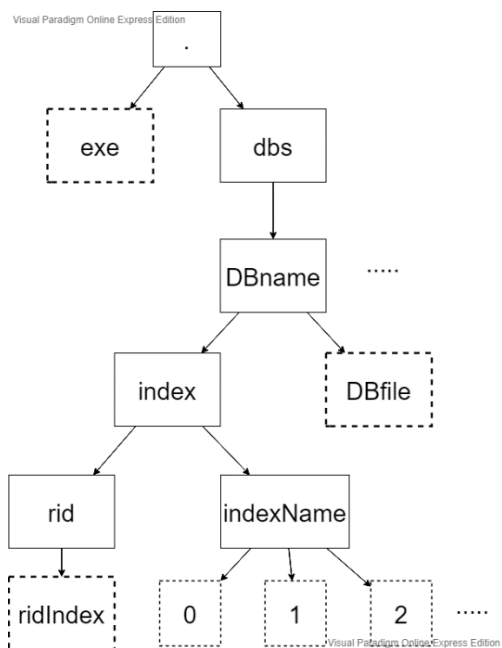
1. 建立一個資料庫，可以有效的進行資料的新增 (單筆的資料新增、多筆的資料匯入)、搜尋、刪除。並加入索引的功能，加速操作的效能。
2. 利用測試資料測試，評估資料庫效能。

## 開發環境和使用工具

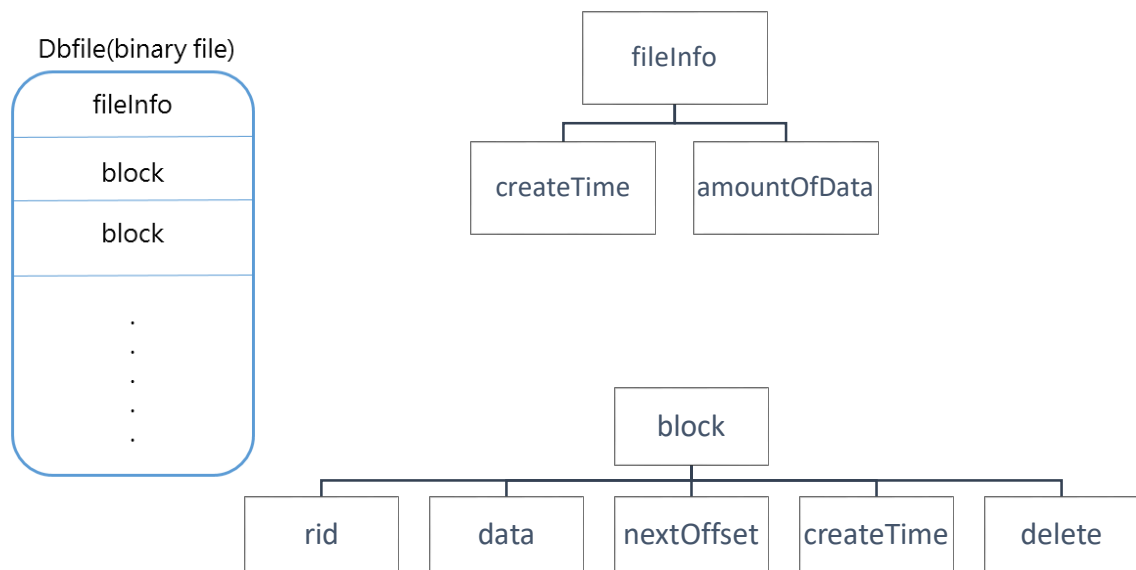
- C 語言
- LinuxMint 19 (Based on Ubuntu 18.04)
- gcc 4.7.0

ps. 資料庫使用 Linux 底層 library，因此目前只能在 Unix-like 環境中執行

## 目錄結構



## 資料儲存



當建立資料庫時會產生二進位檔案 Dbfile 儲存，並初始化檔案中的 fileInfo 資訊，將 amountOfData 歸零，並且儲存目前的時間做為 createTime。

有新的資料進入資料庫的時候會更新 fileInfo 裡面的 amountOfData 並且新增一個 block。block 內的 rid 為一遞增的數值，會在 block 新增時自動給予其相對應的數字；block 內的 data 是負責儲存資料的地方，當一筆資料被匯入時，

資料庫會自動分成一筆或多筆的 key-value，格式皆為字串的形式，存入後格式

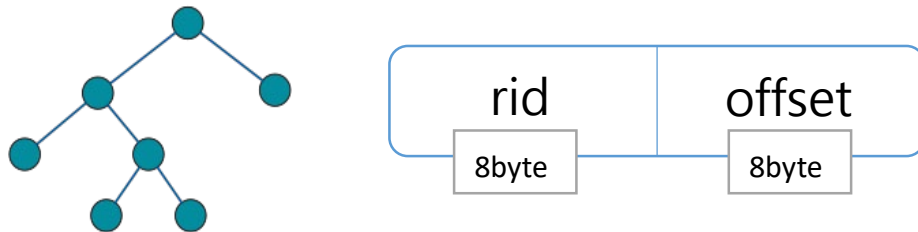
為

```
key1 \0 value1 \0 \0 key2 \0 value2 \0 \0
```

，大小上限是 2KB，當存入資料大於 2KB 時會再自動分配一個 block 儲存，並記錄下一個 block 的位置於 nextOffset 中。block 中也會儲存建立的時間，並有一個 delete 的 tag 紀錄當前資料是否被刪除。

## 索引

### 1. rid – offset

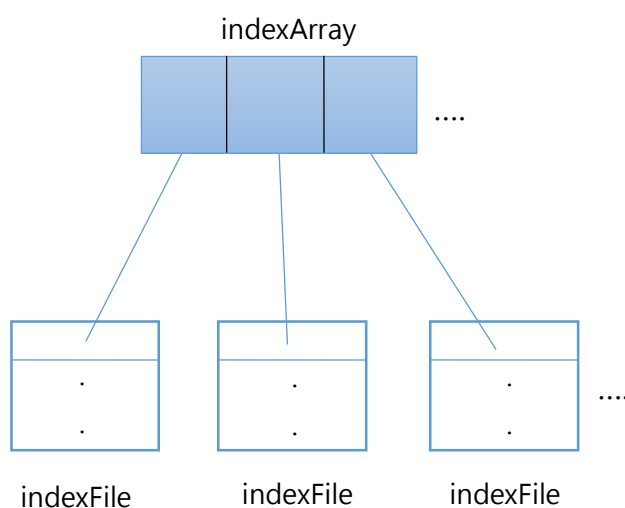


在資料新增/匯入時會自動新建 rid-offset 的索引，並在 database 被使用時自動讀進記憶體中，形成一棵 in-memory BST，在資料庫關閉時寫回硬碟中。

優點：速度快，可以快速的利用 rid 找到需要的資料 (時間複雜度  $O(\log n)$ )

缺點：資料量越大時占用記憶體越大，讀取時間稍久 (9 百萬筆資料讀取時間約 7 秒)

### 2. Sequential index



| rid | indexData | offset |
|-----|-----------|--------|
|-----|-----------|--------|

- indexArray
  - In-memory
- indexFile
  - 8KB
  - In-disk

循序索引。讀取 rid、offset 並將選擇的欄位名稱的 value 存進 indexData 中，接著對 value 們進行排序，依序存入各 indexFile 中，每一個 indexFile 大小為 8KB，空間不夠時創立一個新的 indexFile 繼續儲存。

讀取時，分別讀取各 indexFile 中的第一筆 index 資料進入 indexArray(in-memory)。搜尋時，先依序搜尋 indexArray，如果符合則搜尋成功，如果不符合且 indexArray[i+1] 中的資料大於欲搜尋資料，則進入 indexFile[i] 中進行搜尋，如仍搜尋失敗則無此資料。

時間複雜度： $O(n)$

## 搜尋

如果欲搜尋的欄位有依欄位建立索引則利用索引進行搜尋，如果沒有，則從頭讀取存入的資料，比對欲搜尋的 key-value，輸出所有符合條件的結果。

## 刪除

進行搜尋找到欲刪除的資料(block)並將 block 的 delete tag 修改為 true 完成刪除。

測試

環境

CPU: Intel Core i7 4720HQ @ 2.60GHz

RAM: 8GB

Disk: Micron 120GB SSD

OS: LinuxMint 19 (Based on Ubuntu 18.04)

測試資料

檔案 : youtube.rec

大小 : 2.1GB

數量 : 9312084 筆

格式 :

@url: string

@title: string

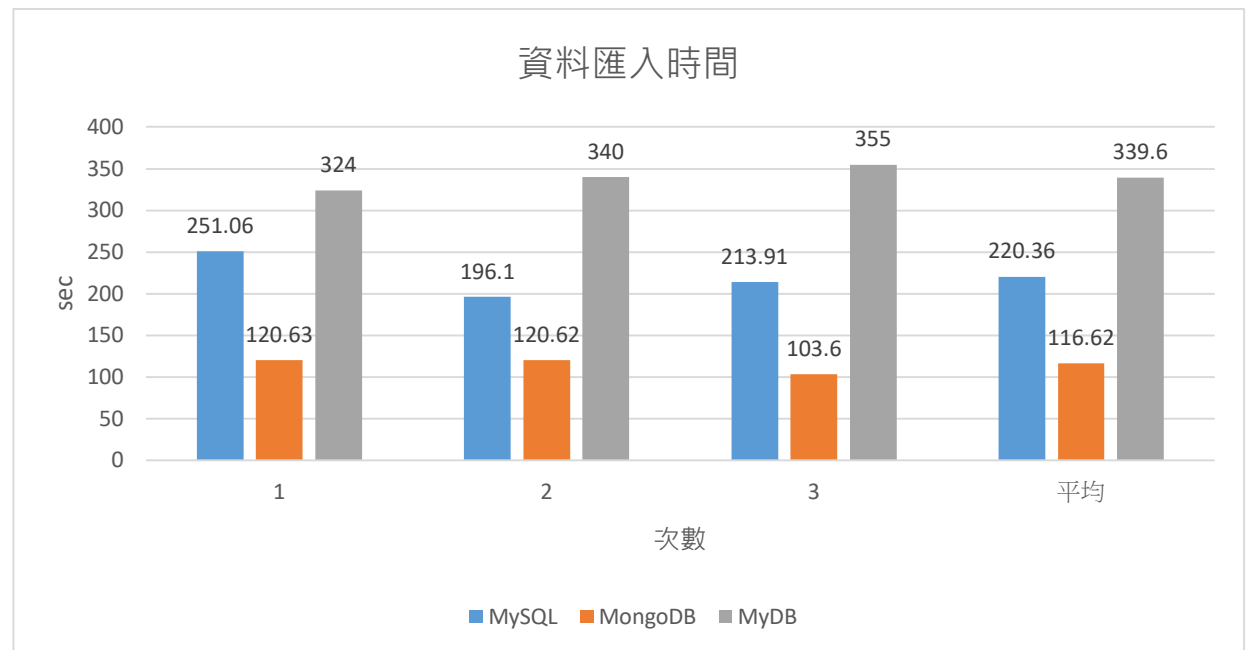
@content: string

@viewCount: string

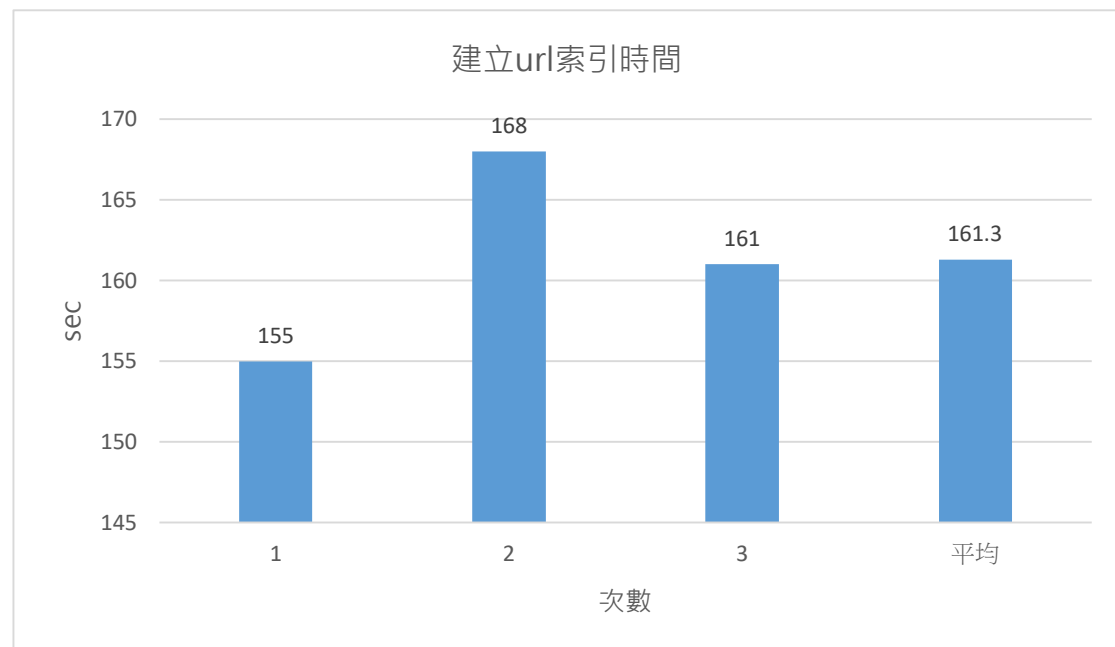
@res: string

@duration: string

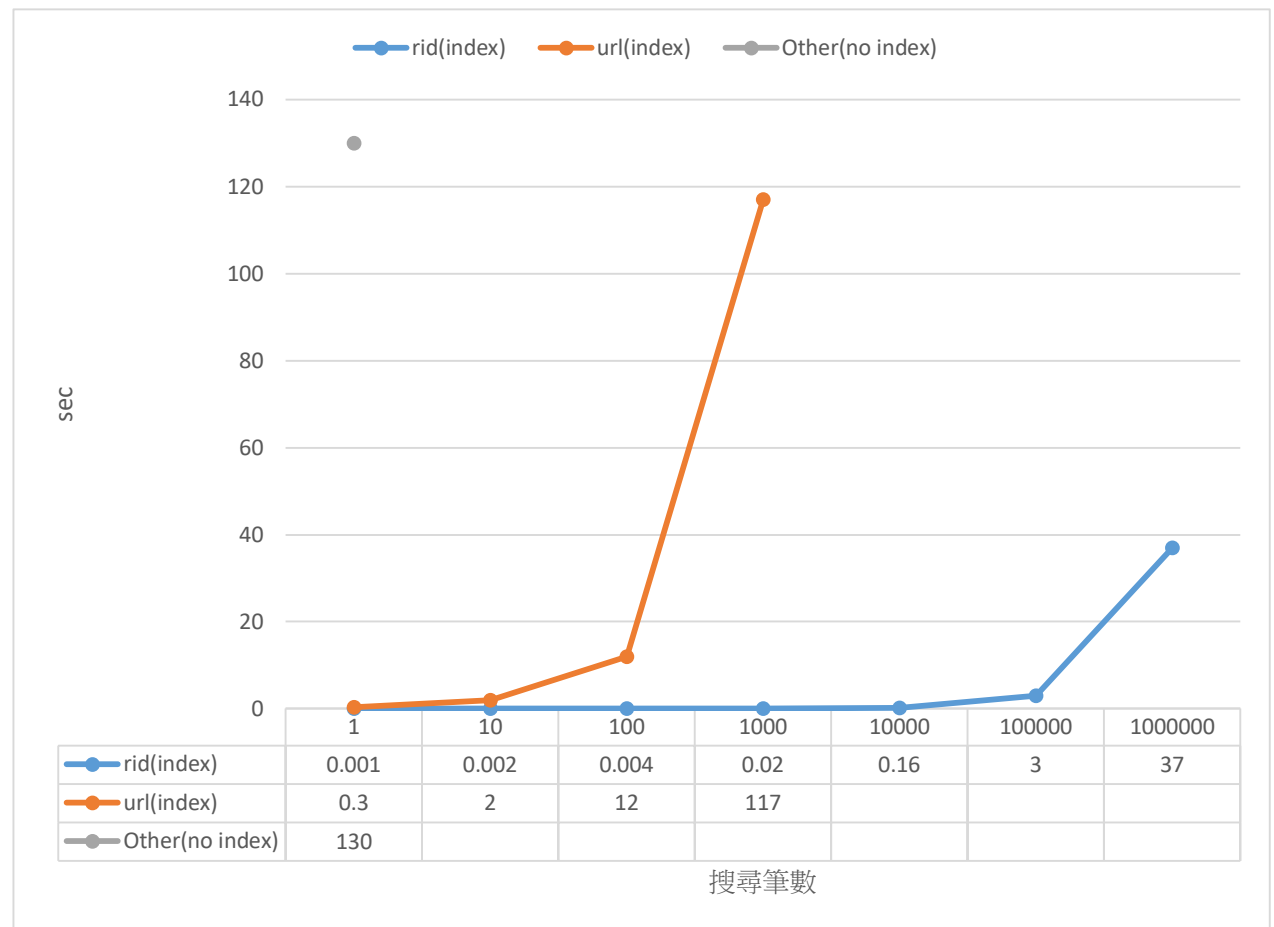
## 資料匯入



## 資料索引



## 資料搜尋



## 資料刪除

|     | 1%(by rid) | 10%(by rid) | 1%(random) | 10%(random) |
|-----|------------|-------------|------------|-------------|
| sec | 37         | 45          | >10min     | x           |

## 附錄

Github : <https://github.com/az801590/myDB>