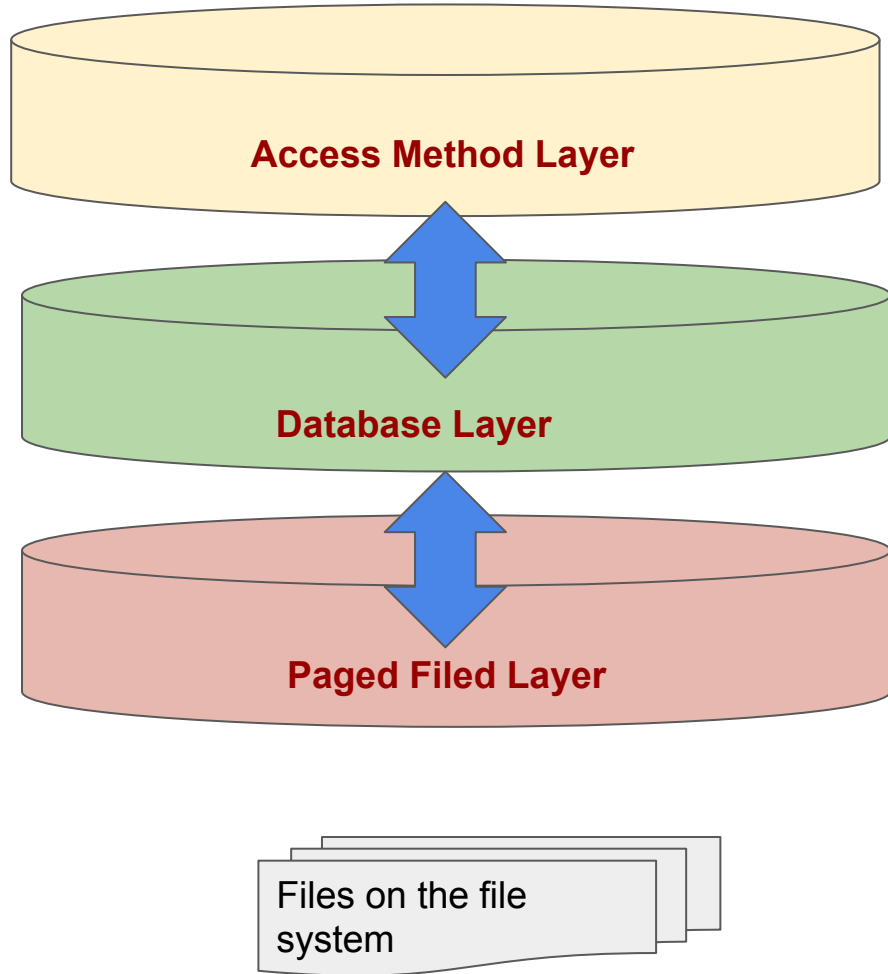


CS 387 Spring 2022

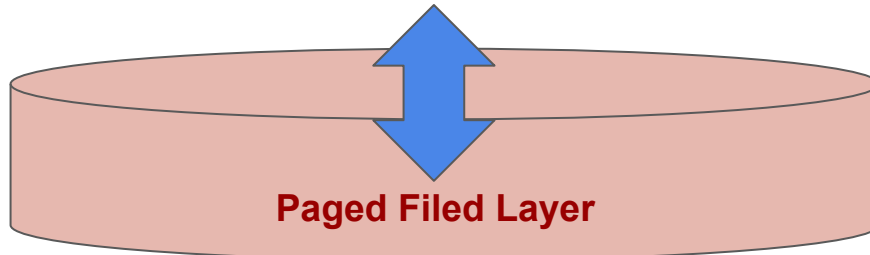
Lab 5 - DB Internals

ToyDB - A Layered Architecture



The PF Layer (pf.pdf)

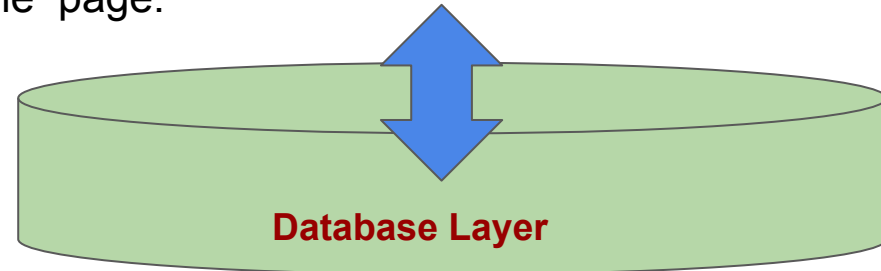
- Allows File I/O in terms of pages
 - These pages are used by upper layers to store indexes as well as data
- API
 - `PF_CreateFile(fileName) & PF_DestroyFile(fileName)`
 - `PF_OpenFile(fileName) & PF_CloseFile(fileName)`
 - `PF_GetFirstPage(fileDes,pageNum,pageBuf)`
 - `PF_GetNextPage(fileDes ,pageNum,pageBuf) & PF_GetThisPage(fileDes ,pageNum,pageBuf)`
 - `PF_AllocPage(fileDes ,pageNum,pageBuf) & PF_DisposePage(fileDes ,pageNum)`
 - `PF_UnfixPage(fileDes ,pageNum,dirty)` (Evicts a page. If marked dirty, writes it back)



The DB Layer (you must write this)

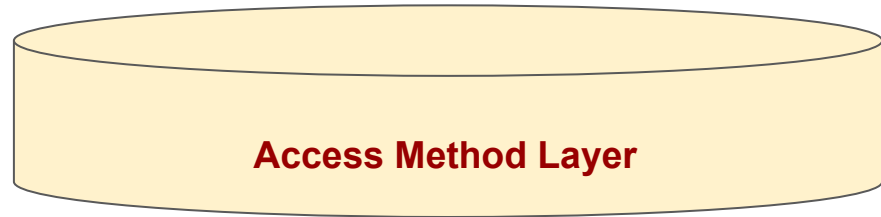
- Converts pages from the PF layer into DB records with records/tuples (containing individual columns)
- What you have to do: structure each page as a slotted-page structure. That is, the header at the top of the page must contain the following information:
 - an array of pointers (offset within the page) to each record,
 - the number of such records, and
 - the pointer to the free space.

The actual record data is stored bottom up from the page.



The AM Layer (am.pdf) - you will build this.

- Exports indexing and DB access methods
 - You must write parts of this to implement a B Tree index and store the given file you will be asked to read into this index.
- Use a paged file from the PF layer to represent a B+ tree index,
- API this will export:
 - `AM_CreateIndex(fileName, indexNo, attrType, attrLength) & AM_DestroyIndex(fileName, indexNo)`
 - `AM_InsertEntry(fileDes, attrType, attrLength, value, recId) & AM_DeleteEntry(fileDes, attrType, attrLength, value, recId)`
 - `AM_OpenIndexScan(fileDes, attrType, attrLength, op, value) & AM_CloseIndexScan(scanDesc)`
 - `AM_FindNextEntry(scanDes)`



Putting it all together

- Successfully load a CSV file given to you using all these APIs (loadddb.c)
- Use the indexes to scan the DB and print out records that satisfy a condition.

