CMPE 321 - Assignment 1
Storage Manager System Design

Neval Tüllük - 2014400216

Summer 2018

1. Introduction

In this assignment, I designed a basic storage manager system. The purpose of this storage manager is to manipulate data efficiently and conveniently. In this design user can create, delete and search for a type, list all of types, create instances of these types, delete those instances, search among them and list all of them which means DML and DLL operations are supported. In system catalogue, the metadata of the system is stored, so the necessary information is supplied to system in system catalogue. In data files, each type in the system catalogue have its own file, named after their type name, and holds the instances of that type.

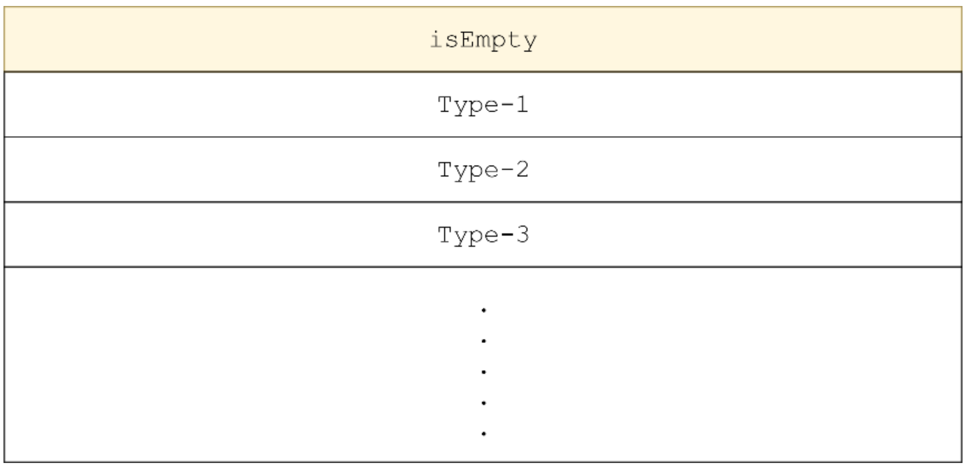
2. Assumptions

- Page size is limited to 1KB.
- A file can contain up to 30 pages.
- The maximum number of fields of a type is 8.
- The minimum number of fields of a type is 1.
- The maximum length of a type name is 16 characters.
- The maximum length of a field name is also 16 characters.
- For each type there is a specific file denoted by the name of that type.
- The record of a type is stored in the file of its name.
- Each type has its unique type name, and the first field name of the type is the primary key.
- A type in the system catalogue has fixed size. (145bytes)
- A record of a type has fixed size.

3. System Design

3.1.System Catalogue

System catalogue is where the metadata of the system. In this project the name of the system catalogue file is “system_cat.txt”. In the header of a system catalogue page the fullness/emptiness of the page is denoted with isEmpty field (1 byte). Then the types are stored and the structure of a type and the system catalogue page is as follows:



The header of a type contains isEmpty field as well (1 byte) to indicate whether the type is deleted before and can be overwritten or not. The types have and the fields in types has the same size, so even if the type has less than 8 fields, the empty fields will be 16 bytes. And if the data (type name or field name) is less than 16 bytes, its completed to 16 bytes. So, the size of a type in system catalogue is fixed and 145 bytes.

isEmpty	type-name	field-name1	field-name2	.....	field-name8

3.2. Data Files

Each type is stored in a data file named “type_name.txt”. Data files contain pages. The size of a page is 1 KB.

In the header of each page there is a “page ID”(4 bytes), “next page pointer” (4 bytes) “numOfFields” the number of the fields of the records in that page (4 bytes) and “remaining space” information (4bytes).

page ID	next page pointer	remaining space	numOfFields
record-1			
record-2			
record-3			
. . . .			

Every record has a record header and at least one field. In the record header there is isEmpty information to indicate whether a record has been deleted and can be overwritten or not.

isEmpty	field-1 value	field-2 value	field-3 value	.....

4. Operations

4.1. DLL Operations

4.1.1. Create a Type

```
begin
     get typeName from user
     get numberOfTypes from user
     open system_cat.txt
     for (types in system_cat.txt)
           get isEmpty field from type header
           if (the isEmpty field of the type is true)
                 rewrite typeName
                 go to first field_name
                 for (numberOfTypes)
                       get field_name from user
                       rewrite field_name
                       go to next field_name
                 end for
                 for (maxNumberOfTypes - numberOfTypes)
                       set field_name to null
                       go to next field_name
                 end for
                 make isEmpty false
                 break
           else go to next type
           end if
     end for
     create empty type array
     for (numberOfTypes)
           get field_name from user
           fill the field_name
           go to next field_name
     end for
end
```

4.1.2. Delete a Type

```
begin
     get typeName from user
     open system_cat.txt
     for (types in system_cat.txt)
           if (type name of the record equals to typeName)
                 make isEmpty field in header of that type true
           else go to next type
           end if
     end for
end
```

4.1.3. List All Types

```
begin
     open system_cat.txt
     for(types in system_cat.txt)
          if(isEmpty field in type header is false)
               print typeName
               go to next type
          end if
     end for
end
```

4.2. DML Operations

4.2.1. Create a Record

```
begin
     get typeName from user
     open typeName.txt
     for(pages in typeName.txt)
          get remainingSpace from page header
          if(remainingSpace is equals to 0)
          go to next page
          else
               get numOfFields
               for (records in current page)
                    if (the isEmpty field of the record is
                true)
                         go to first field
                         for (numOfFields)
                              get recordData from user
                              rewrite recordData
                              go to next field
                         end for
                         make isEmpty false
                         break
                    else go to next record
                    end if
               end for
               create new empty record array
               for (numOfFields)
                    get recordData from user
                    fill recordData
                    go to next field
               end for
               break
          end if
     end for
end
```

4.2.2. Delete a Record

```
begin
     get typeName from user
     open typeName.txt
     get primaryKey from user
     for (records in typeName.txt)
          if (primary key of the record equals to primaryKey)
               make isEmpty of that record true
               update remainingSpace in the page header
          else go to next record
          end if
     end for
end
```

4.2.3. Search for a Record

```
begin
     get typeName from user
     open typeName.txt
     get primaryKey from user
     for (records in typeName.txt)
          if (primary key of the record equals to primaryKey &
isEmpty is false)
               print record
          else go to next record
          end if
     end for
end
```

4.2.4. List All Records of a Type

```
begin
     get typeName from user
     open typeName.txt
     for (records in typeName.txt)
          if(isEmpty of this record is false)
               print record
               go to next record
          end if
end for
```

5. Conclusions and Assessment

In this project I have tried to design a simple storage manager, with given assumptions I think that this system supposed to work fine. With this project I understood the main points about a database management system and how it should work and I tried to keep my design simple. There are some inefficiencies about storage space, for example to simplify and fasten the delete and insert mechanisms all the types have fixed number of fields (even if the user specifies a type with two fields the 6 fields remains and considered null), and the type names and the field values kept 16 characters (16 bytes) even if there are only two characters for that attribute.

After doing this project I think I understood the database management systems and their goals better. I believe the things I have discovered during this project will help me develop better solutions to database related problems.