

SimpleDB Lab Report

Introduction

The purpose of this lab is to implement the foundational components of a simple database management system (DBMS) called SimpleDB. This lab focuses on the core elements required for a basic DBMS, including tuple and table management, a catalog for storing metadata, a buffer pool for managing memory-resident pages, and a heap file access method.

Exercise 1: Tuple and TupleDesc

Implementation

The first task was to implement the `TupleDesc` and `Tuple` classes. `TupleDesc` describes the schema of a tuple, including the types and names of its fields. The `Tuple` class represents a single record in a table, containing data for each field as specified by its `TupleDesc`.

Key Challenges and Solutions

- Ensuring that `TupleDesc` correctly encapsulates the schema information, including field types and names.
- Implementing `Tuple` to interact with `TupleDesc` and manage field values effectively.

Exercise 2: Catalog Implementation

The `Catalog` class maintains a list of tables and their schemas in the database. It allows for adding new tables and retrieving information about existing tables.

Key Challenges and Solutions

- Designing a data structure that efficiently stores and retrieves table metadata.
- Ensuring that table names and schemas are handled correctly, with proper conflict resolution for naming.

Exercise 3: BufferPool Implementation

The `BufferPool` class is responsible for caching pages in memory to improve disk access times. For this lab, only the constructor and `getPage` method were required, with a simple exception throw for exceeding the buffer capacity.

Key Challenges and Solutions

- Understanding the role of the buffer pool in the overall system and how it interacts with other components.
- Implementing a basic page caching mechanism without an eviction policy, focusing on error handling for capacity overflow.

Exercise 4: HeapFile Access Method

Heap files are a simple way to store unsorted data on disk. The `HeapPage`, `HeapPageId`, and `RecordID` classes represent the pages, identifiers, and record IDs within these files, respectively.

Key Challenges and Solutions

- Grasping the concept of heap files and how they store tuples with a bitmap header indicating valid slots.
- Implementing methods to manage and iterate over tuples within a heap page, including bit manipulation for header flags.

Exercise 5: HeapFile Implementation

Building on the previous exercise, this task involved calculating the number of pages in a heap file and reading pages from disk.

Key Challenges and Solutions

- Calculating the correct offset for reading pages from disk, considering the page structure.
- Implementing an iterator for the `HeapFile` that retrieves tuples from disk through the buffer pool.

Exercise 6: Operators

Operators in SimpleDB execute the operations of the relational algebra. The `SeqScan` operator was implemented to sequentially scan tuples from a table.

Key Challenges and Solutions

- Understanding the iterator-based approach for operator execution and how to implement it for a SeqScan operator.
- Ensuring that the SeqScan operator correctly interacts with the HeapFile and BufferPool to retrieve tuples.

Conclusion

This lab laid the groundwork for a simple DBMS by implementing core components such as tuples, tables, a catalog, a buffer pool, and a basic heap file access method. The challenges encountered involved understanding the underlying concepts and ensuring proper interaction between the various components. The successful completion of this lab sets the stage for further development and implementation of more complex features in subsequent labs.