Indexes

Useful SQLite Commands

- *.eqp on|off*
  - Show execution plan for SQL query
- *.timer on|off*
  - Show execution time for SQL statement
- *.output file*
  - Print SQL statement output to *file*
- *.read file*
  - Execute SQL statements from *file*

Query Types

- Full table scan
  - select l_orderkey from lineitem
- Point query
  - select l_orderkey from lineitem where **l_quantity = 10**
- Range query
  - select l_orderkey from lineitem where **l_quantity < 10**
  - select l_orderkey from lineitem where **l_quantity >= 10 and l_quantity <= 20**

Query Execution Plans

- Full table scan
  - select l_orderkey from lineitem
    - 60,175 tuples
    - **`--SCAN TABLE lineitem**
- Point query
  - select l_orderkey from lineitem where l_quantity = 10
    - 1,182 tuples
    - **`--SCAN TABLE lineitem**
- Range query
  - select l_orderkey from lineitem where l_quantity < 10
    - 10,816 tuples
    - **`--SCAN TABLE lineitem**
  - select l_orderkey from lineitem where l_quantity >= 10 and l_quantity <= 20
    - 13,071 tuples
    - **`--SCAN TABLE lineitem**

Indexes

- Query time is proportional with the number of tuples accessed by the query
  - More tuples accessed → larger query time
- Reduce number of accessed tuples by creating a copy of an attribute and sort it increasingly
  - Binary search on sorted data
  - Pointer to the complete tuple
- Trade-off space for query time

Indexes in SQLite

- **CREATE INDEX *lineitem_idx_l_quantity* ON lineitem(l_quantity)**
- **DROP INDEX *lineitem_idx_l_quantity***
- Database server decides when and how to use indexes for query processing
  - User cannot control index usage

Query Execution Plans with Indexes

- Full table scan
  - select l_orderkey from lineitem
    - 60,175 tuples
    - **`--SCAN TABLE lineitem**
- Point query
  - select l_orderkey from lineitem where l_quantity = 10
    - 1,182 tuples
    - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity=?)**
- Range query
  - select l_orderkey from lineitem where l_quantity < 10
    - 10,816 tuples
    - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity<?)**
  - select l_orderkey from lineitem where l_quantity >= 10 and l_quantity <= 20
    - 13,071 tuples
    - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity>? AND l_quantity<?)**

Database Size Increase

- *ls -la* command
- Before CREATE INDEX
  - data/tpch.sqlite: **11288576 bytes**
- After CREATE INDEX
  - data/tpch.sqlite: **11862016 bytes**
  - Increase of **573440 bytes**

Query Execution Time (Decrease)

- select l_orderkey from lineitem
  - **`--SCAN TABLE lineitem** → **14 ms**
- select l_orderkey from lineitem where l_quantity = 10
  - **`--SCAN TABLE lineitem** → **6 ms**
  - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity=?)** → **3 ms**
- select l_orderkey from lineitem where l_quantity < 10
  - **`--SCAN TABLE lineitem** → **8 ms**
  - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity<?)** → **16 ms**
- select l_orderkey from lineitem where l_quantity >= 10 and l_quantity <= 20
  - **`--SCAN TABLE lineitem** → **8 ms**
  - **`--SEARCH TABLE lineitem USING INDEX ind_l_quantity (l_quantity>? AND l_quantity<?)** → **19 ms**

INSERT Execution Time Increase

- **insert into lineitem (select * from lineitem)**
  - No index → **120 ms**
  - Index → **156 ms**

Index Recommendation

- SQLite recommends indexes for a query based on data and existing indexes

```
.expert
select l_orderkey from lineitem where l_quantity = 10
```

- CREATE INDEX lineitem_idx_l_quantity ON lineitem(l_quantity)

Indexes Summary

- Increase in storage space
- Decrease in query execution time
  - Only for very selective queries with result tuples very small compared to table tuples (1,182/60,175)
- Increase in MODIFICATION (I/U/D) execution time
  - Modify both the table and the index