Physical Design

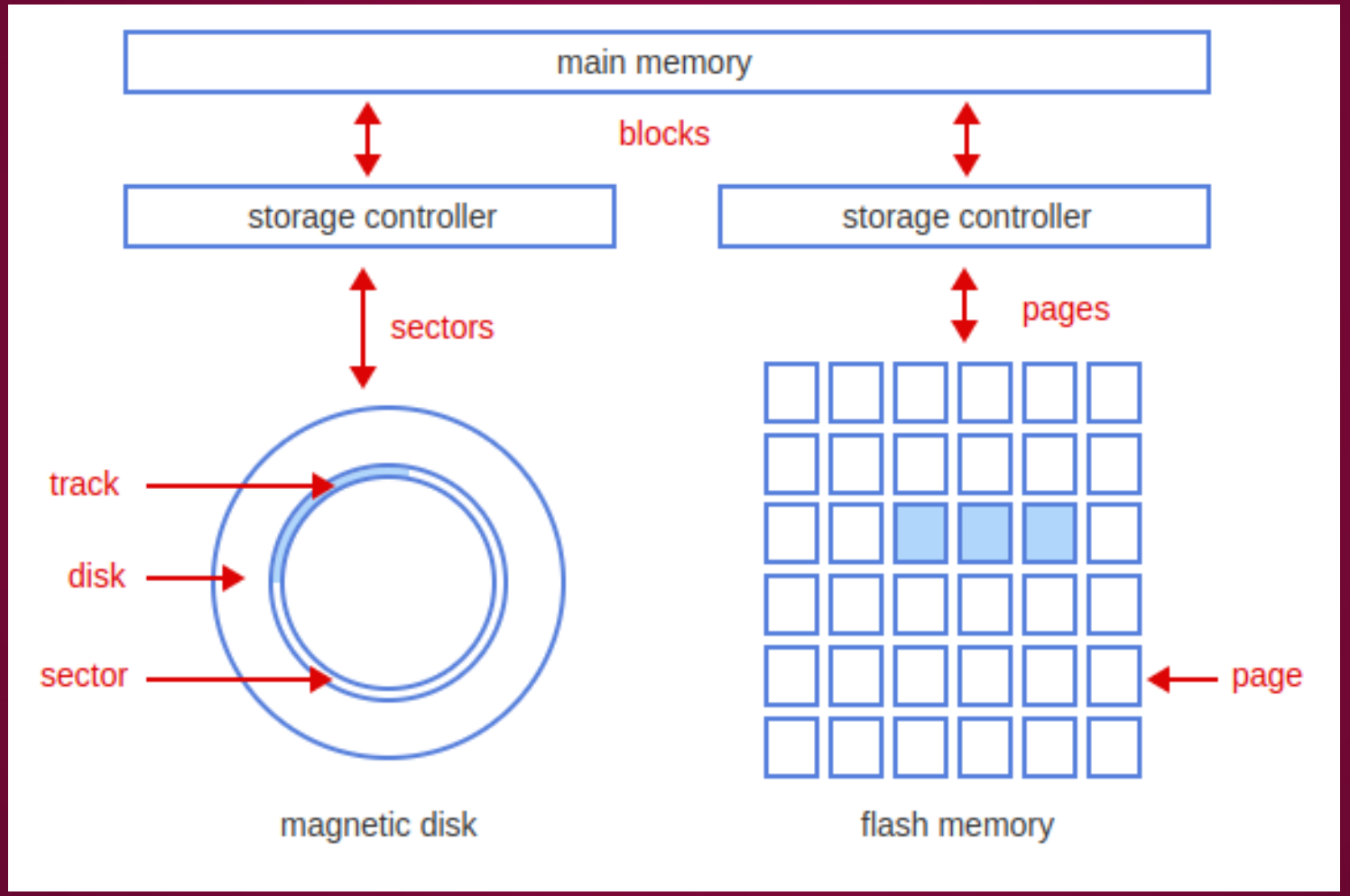
UNIVERSITY OF MONTANA

Working in the background

- Decisions affect performance but never affect query results

UNIVERSITY OF MONTANA

Storage Media



Data Organization

- Row oriented
  - Better for transactional applications
- Column oriented
  - Better for analytical applications

Table Structure

- Heap - No order is imposed
  - Fast write, slow read
- Sorted table - Rows are sorted by a column
  - Fast read, slow write
- Hash table - Data is grouped together based on hash
  - Fast for individual reads/writes, slow for bulk
- Table cluster - Multiple tables interleaved
  - Specialized for a particular join, slow for most other operations

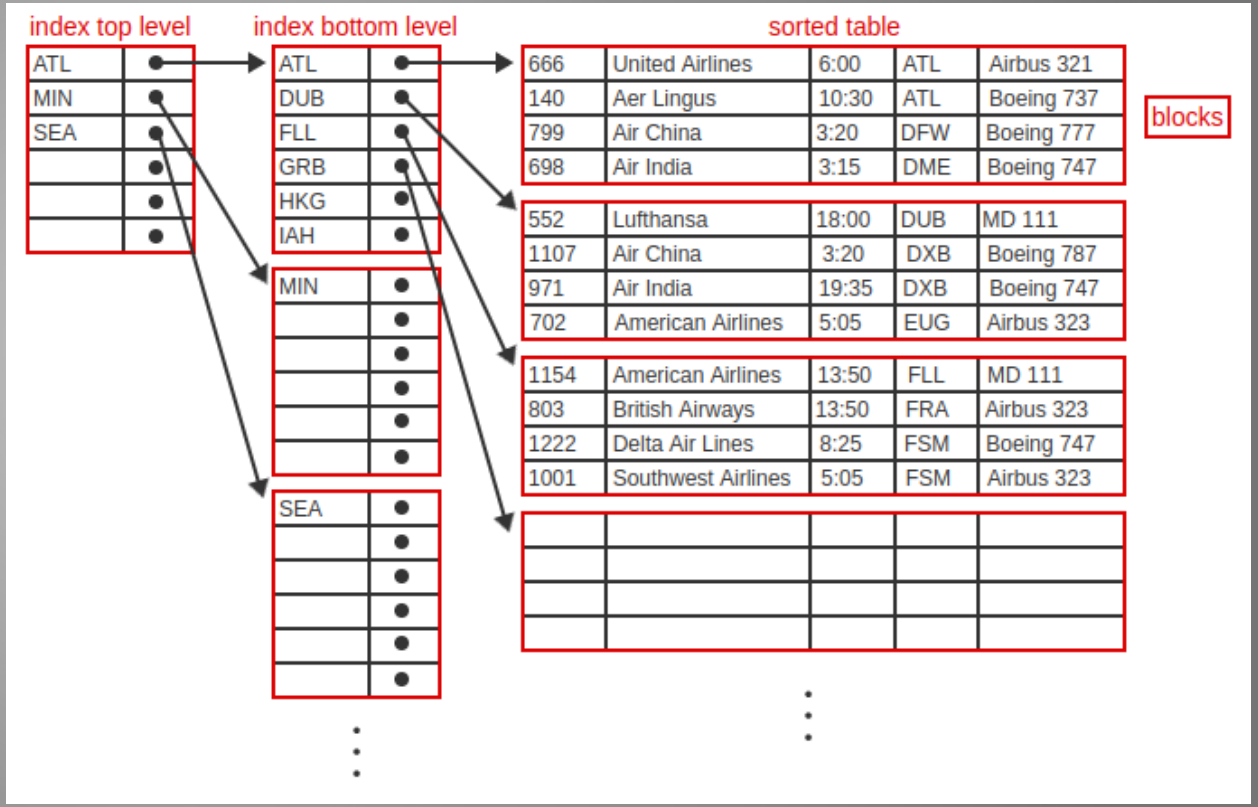
Indexes

- Addresses the main weakness of a heap table
- One column (along with a pointer) is sorted
- Allow for the possibility of a binary search
- Multiple indexes?
- Dense vs Sparse

Socratica Video

Multi-level Indexes

- B+tree index
  - Values are repeated in lower level indexes
- B-tree index
  - Values are not repeated in lower level indexes



Binary Search - Find 69951757

4055505
6995630
9086405
9270322
9362777
10484970
11848518
12622330
13440190
13689200
14082277
15347211
15419461
16483496
18521572
19870627
20138740
20406038
21666247
23408187
25419539
28001595
28867473
29515064
30139471
32853477
32964207
33657355
33671279
34096808
35700734
36312160
39128058
40095578
42240036
42942475
43247718
43541621
49147525
51034515
51615167
54182046
54458130
55967518
56678759
58751875
60528887
61862972
62360432
63236935
63420879
69951757
70295119
71129543
71872482
72060865
74782400
76091035
76327014
76337118
77100229
77193195
79727269
80468817
80769463
86354069
89028599
89965459
92439732
92957685
93201595
93510255
93696884
95438865
95628831
96039351
96349184
99070162
99569182
99945508

UNIVERSITY OF MONTANA

Storage Engine

- InnoDB - Transactional DB (Default)
  - Heap or sorted tables
  - B+tree indexes
- MyISAM - Analytical applications with limited updates
  - B+tree indexes
- MEMORY - all data is stored in memory
  - B+tree or hash indexes

SQL for working with indexes

- CREATE INDEX IndexName on TableName (Column1, Column2, etc);
- DROP INDEX IndexName on TableName
- SHOW INDEX FROM TableName;

EXPLAIN Statement

- Shows how an CRUD statement will be executed
- Help with deciding on an index
- Identify slow queries

```
EXPLAIN
  SELECT FlightNumber
  FROM Flight
  WHERE AirportCode IN
      (SELECT AirportCode
      FROM Airport
      WHERE Country = "Cuba");
```

UNIVERSITY OF MONTANA