B-Tree

To learn about B-Tree index I’ve read 2 references and I found the second one more helpful.

1. Main idea of B-tree index is to minimize the number of physical reads. Since the data structure is sorted, B-tree index can be used effectively for range scans. Seeks are not so effective compared to hash indexes. Deletes cause fragmentation, sequential inserts are relative cheep, non-sequential inserts and updates may be very expensive; however, modern RDMS usually handle them with decent performance.

Reference : <https://dev.mysql.com/doc/refman/5.1/en/mysql-indexes.html>

1. Indexes in MySql works like an index in a book. While, indexes in a book tell you about the pages on which a term occurs, indexes in MySql tell you the rows that contain the matching data. An index contains values from one or more columns in a table. If you index more than one column, the column order is very important, because MySQL can only search efficiently on a leftmost prefix of the index. Creating an index on two columns is not the same as creating two separate single-column indexes. In this post, I will discuss about B-Tree index and it’s working in MySql.

**B-Tree Indexes**

This is the default index for most storage engines in MySql. The general idea of a B-Tree is that all the values are stored in order, and each leaf page is the same distance from the root.

A B-Tree index speeds up data access because the storage engine doesn’t have to scan the whole table to find the desired data. Instead, it starts at the root node. The slots in the root node hold pointers to child nodes, and the storage engine follows these pointers. It finds the right pointer by looking at the values in the node pages, which define the upper and lower bounds of the values in the child nodes. Eventually, the storage engine either determines that the desired value doesn’t exist or successfully reaches a leaf page. Leaf pages are special, because they have pointers to the indexed data instead of pointers to other pages.

Reference: <https://guptavikas.wordpress.com/2012/12/17/b-tree-index-in-mysql/?blogsub=confirming#subscribe-blog>