

B+-Tree Example

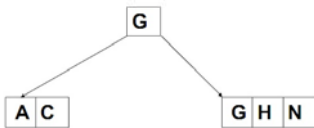
Harshilsurti
<http://www.scribd.com/doc/18211/B-TREE-TUTORIAL-PPT>

Creating a B+ Tree

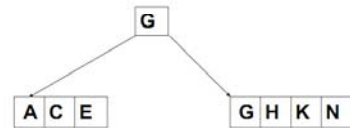
C N G A H E K Q M F W L T Z D P R X Y S

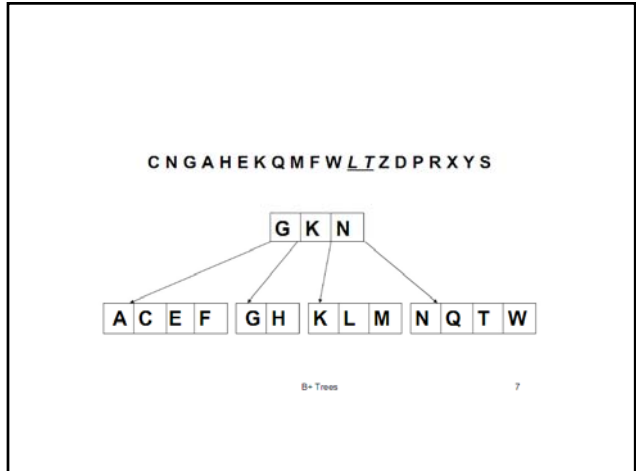
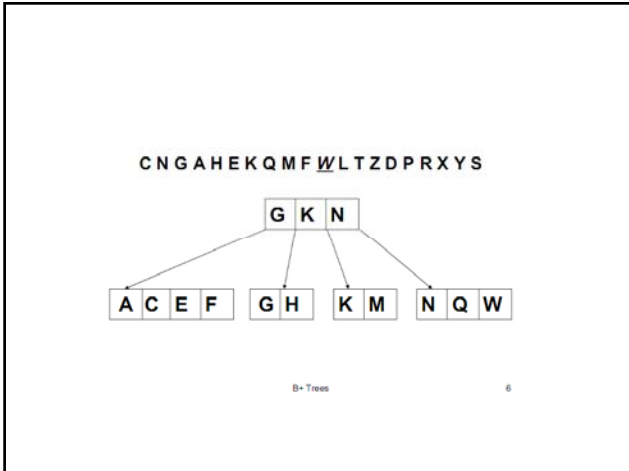
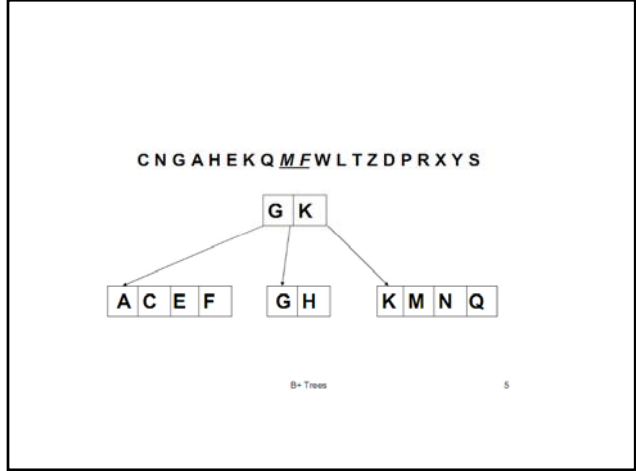
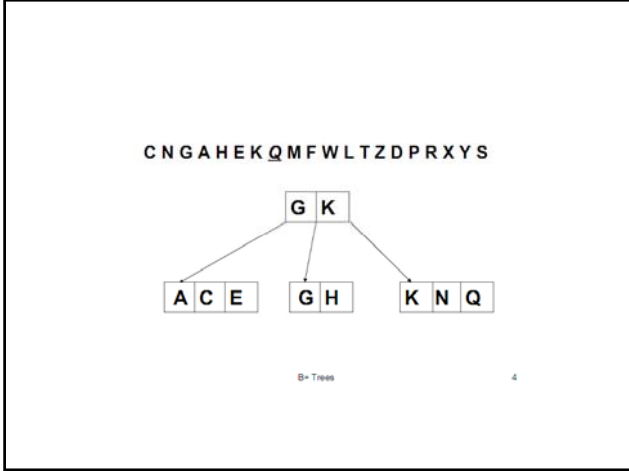
A C G N

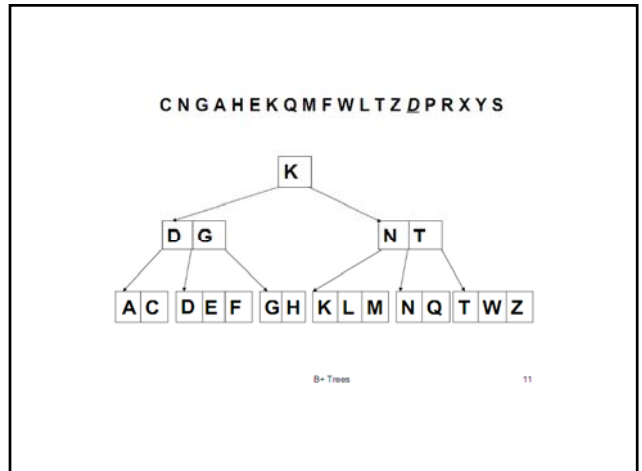
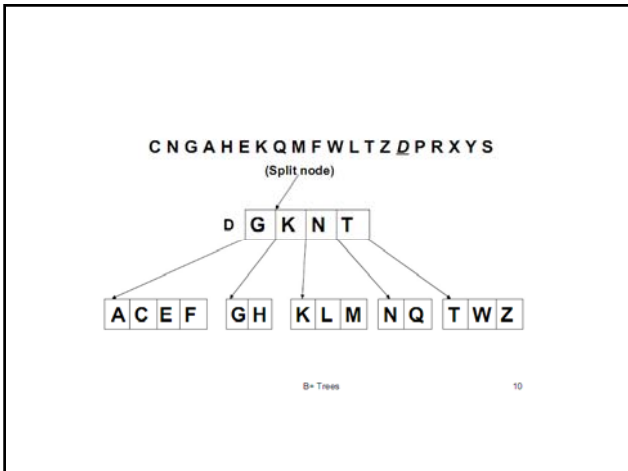
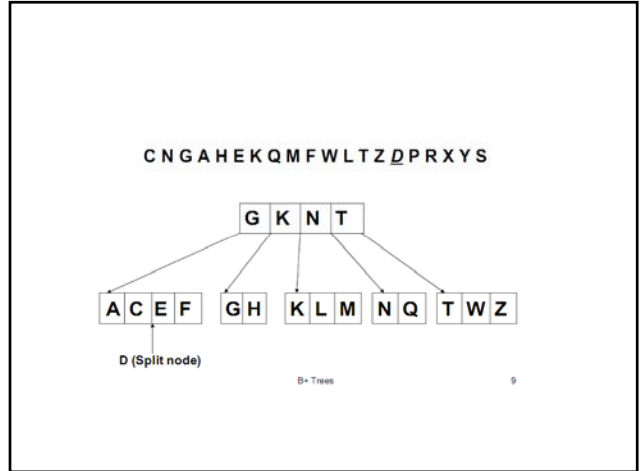
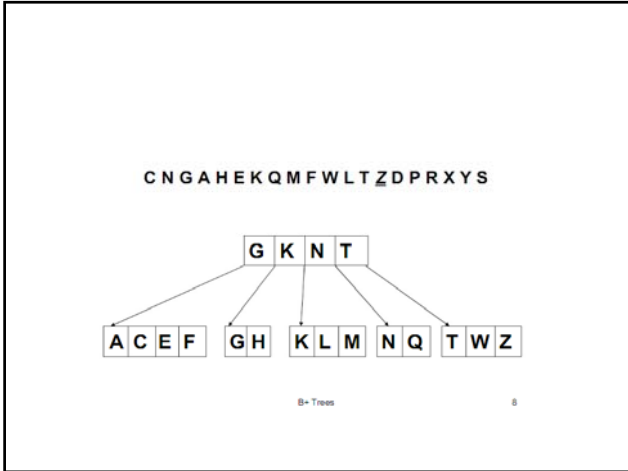
C N G A H E K Q M F W L T Z D P R X Y S

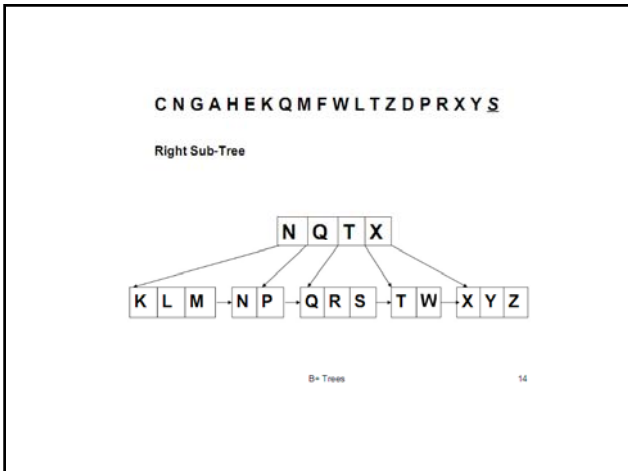
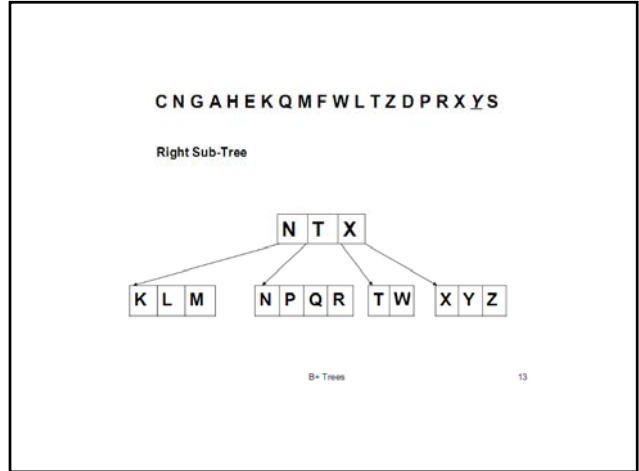
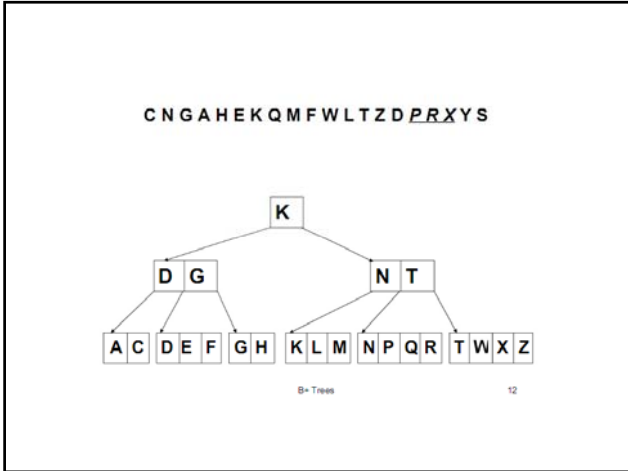


C N G A H E K Q M F W L T Z D P R X Y S









Speed in B+ Tree Index

- In processing a query, we traverse a path from the root to a leaf node. If there are K search key values in the file, this path is no longer than $\log_{(n/2)} K$, where n is number of links possible in any given node.
- This means that the path is not long, even in large files. For a 4k byte disk block with a search-key size of 12 bytes and a disk pointer of 8 bytes, n is around 200. If $n = 100$, a look-up of 1 million search-key values may take $\log_{100}(1,000,000) = 4$ nodes to be accessed. Since root is usually in the buffer, so typically it takes only 3 or fewer disk reads.

B+ Trees 15