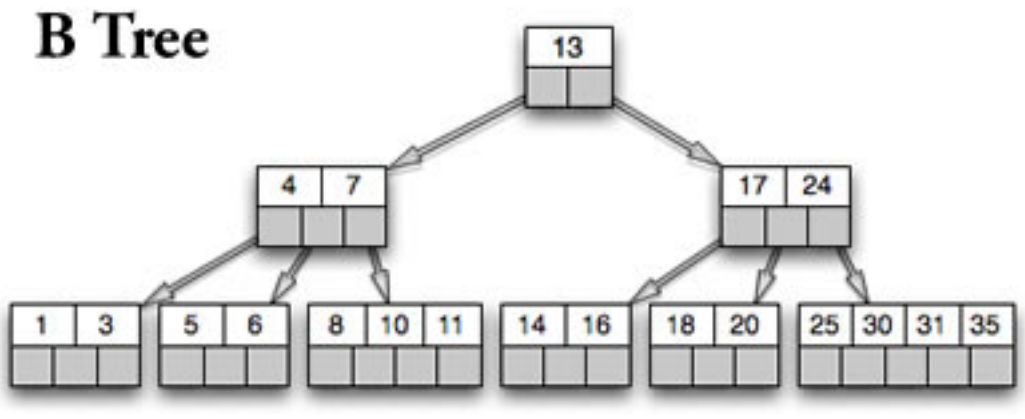
DB

Difference Between | Descriptive Analysis and Comparisons

Difference between B Tree and B+ Tree

Key difference: In computers, the binary trees are tree data structures that store the data, and allow the user to access, search, insert and delete the data at the algorithmic time. The difference between a B and B+ tree is that, in a B-tree, the keys and data can be stored in both the internal and leaf nodes, whereas in a B+ tree, the data and keys can only be stored in the leaf nodes.



The Binary trees are balanced search trees that are designed to work well on direct access secondary storage devices such as magnetic disks. Rudolf Bayer and Ed McCreight invented the concept of a B-tree.

A B-tree is a generalized binary search tree, in which any node can have more than two children. Each internal node in a B-tree contains a number of keys. These keys separate the values, and further forms the sub-trees. The internal nodes in a B-tree can have variable numbers of child nodes, which are arranged within a pre-defined range. At the time when any

Not Found

The requested URL /index.php was
this server.

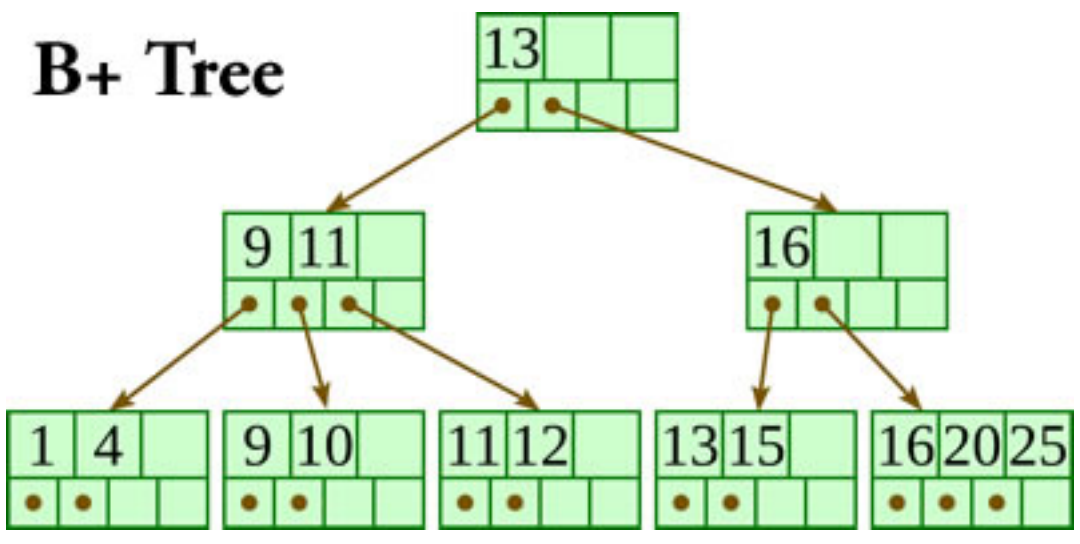
Additionally, a 404 Not Found erro
encountered while trying to use an
to handle the request.

data is inserted or removed from any respective node, there is a change in the number of child nodes. In order to maintain the pre-defined range, internal nodes may be joined or split. In a B-tree, a range of child nodes are permitted, due to which the pre-defined range have to be maintained.

The B-trees do not need to be rebalanced frequently unlike other self-balancing search trees. The nodes in these trees are not always full; hence, the spaces are consumed unnecessary in these trees leading to wastage of space. Only the lower and upper bounds on the number of child nodes are typically fixed for a particular implementation. For example, in a 2-3 B-tree (often simply referred to as a 2-3 tree), each internal node may have only 2 or 3 child nodes.

Additionally, the B-tree is optimized for systems that read and write large blocks of data. It is commonly used in databases and file-systems. In the B tree, all the nodes are kept at the same balancing depths from the root nodes. These depths increase slowly as the number of elements increase; this results in all leaf nodes being one more node farther away from the root. Furthermore, the B- trees are more advantageous when compared to other implementations in the regard to the time take to access the data.

A B+ tree is an n-array tree with a node, which consists of a large number of children per node. The root may be a leaf or a node that contains more than two children. A B+ tree consists of a root, internal nodes and leaves.



A B+ tree is the same as a B tree; the only difference is that, in

Newly Added Differences

- Node.js vs Angular.js
- Real Diamond vs Fake Diamond
- Scared vs Afraid
- Drying vs Dehydration
- Anxious vs Eager
- Heading vs Headline
- Concern vs Worry
- Separation vs Extraction
- Nike vs Adidas
- Professional vs Amateur

Most Popular Differences

- Management vs Administration
- Data vs Information
- Yoghurt vs Curd
- Android vs Windows Phone
- SUV vs MUV vs XUV
- Tablet vs Smartphone
- Taekwondo vs Karate
- Inotropic vs Chronotropic vs Dromotropic

Random Pick

- Continental vs Chinese Food
- NASA vs ISRO
- Pretty vs Gorgeous
- Castor Oil vs Coconut Oil
- Ambience vs Ambiance
- Different Types of Fire

the B+ tree there is an additional level added at the bottom with linked leaves. Also, unlike the B tree, each node in a B+ tree contains only keys and not key–value pairs.

Additionally, the balancing factor or the order of a B+ tree measures the capacity for the internal nodes in a tree, i.e. the number of nodes they can have. The actual number of children for a node is limited for internal nodes. The root is however an exception as it is allowed to have more than two number of children. For example, if the order of a B+ tree is 7, each internal node (except for the root) may have between 4 and 7 children; while the root may have between 2 and 7. The primary value of the B+ tree is in storing data for efficient retrieval in a block-oriented storage context and in particular file-systems.

The primary value of the B+ tree is in storing and maintaining the data, so that the data is not lost. This approach is especially applied in block-oriented storage context and in some particular file-systems. The leaves, which are the bottom-most index blocks, of the B+ tree are often linked to one another in a linked list; hence this makes range queries or an ordered iteration through the blocks simpler and more efficient. Furthermore, the space factor is not wasted in B+ trees. The B+ tree provides an efficient housing data structure format, which makes them simple in accessing and storing. The B+ trees are particularly useful as a database system index, where the data typically resides on a disk.

Comparison between B Tree and B+ Tree:

	B Tree	**B+ Tree**
	A B tree is an organizational structure for information storage and retrieval in the	B+ tree is an n-array tree with a variable but often large number of

Short web descriptions	form of a tree in which all terminal nodes are at the same distance from the base, and all non-terminal nodes have between n and 2 n sub-trees or pointers (where n is an integer).	children per node. A B+ tree consists of a root, internal nodes and leaves. The root may be either a leaf or a node with two or more children.
Also known as	Balanced tree.	B plus tree.
Space	O(n)	O(n)
Search	O(log n)	$O(\log_b n)$
Insert	O(log n)	$O(\log_b n)$
Delete	O(log n)	$O(\log_b n)$
Storage	In a B tree, search keys and data stored in internal or leaf nodes.	In a B+ tree, data stored only in leaf nodes.
Data	The leaf nodes of the three store pointers to records rather than actual records.	The leaf nodes of the tree stores the actual record rather than pointers to records.

Space	These trees waste space	There trees do not waste space.
Function of leaf nodes	In B tree, the leaf node cannot store using linked list.	In B+ tree, leaf node data are ordered in a sequential linked list.
Searching	Here, searching becomes difficult in B- tree as data cannot be found in the leaf node.	Here, searching of any data in a B+ tree is very easy because all data is found in leaf nodes.
Search accessibility	Here in B tree the search is not that easy as compared to a B+ tree.	Here in B+ tree the searching becomes easy.
Redundant key	They do not store redundant search key.	They store redundant search key.
Applications	They are an older version and are not that advantageous as compared to the B+ trees.	Many database system implementers prefer the structural simplicity of a B+ tree.

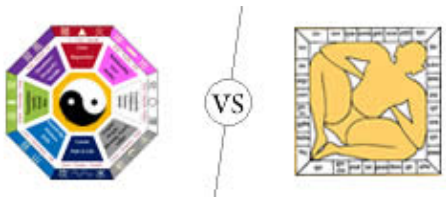
Image Courtesy: 20bits.com, ozark.hendrix.edu

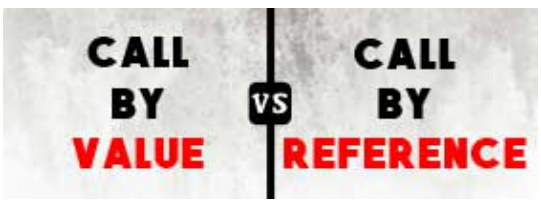
Top 10 Most Searched Differences

Most Searched in Computers & Internets

Most Searched in Pregnancy & Parenting

Most Searched Business & Finance





Add new comment

Comments

B-tree is not binary tree. Binary tree has at most only two leaves per node, thus the name binary.

Janis

Mon, 04/03/2017 - 22:49

reply

many wrong concept in it.So DO NOT FOLLOW THIS FULLY.

Sourav

Mon, 03/06/2017 - 23:32

reply

B tree is not binary tree, it denotes balanced tree

Anonymous

Sat, 11/12/2016 - 02:25

reply

This website is great source of info...:)

Kumar sanu

Tue, 09/06/2016 - 12:41

reply

Valuable info. Lucky me I found your website accidentally, and I am shocked why this coincidence didn't took place earlier! I bookmarked it. tiendas online de ropa interior http://www.infopolis.es/lis.asp?so=tiendas-online-de-ropa-interior

tiendas online ...

Fri, 05/13/2016 - 22:26

reply

This article is very helpful to me as it has full information about b-tree and b+ tree.

Aishwarya

Thu, 05/07/2015 - 12:02

reply

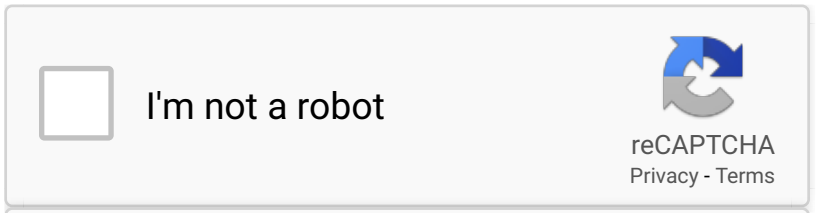
Add new comment

Your name

Comment *

CAPTCHA

This question is for testing whether or not you are a human visitor and to prevent automated spam submissions.



Save