Compact Index File Structure (.idx)

Visual Studio .NET 2003

Compact Index Header Record

Byte offset	**Description**
00 – 03	Pointer to root node
04 – 07	Pointer to free node list (-1 if not present)
08 – 11	Reserved for internal use
12 – 13	Length of key
14	Index options (any of the following numeric values or their sums): 1 – a unique index 8 – index has FOR clause 32 – compact index format 64 – compound index header
15	Index signature
16 – 19	Reserved for internal use
20 – 23	Reserved for internal use
24 – 27	Reserved for internal use
28 – 31	Reserved for internal use
32 – 35	Reserved for internal use
36 – 501	Reserved for internal use
502 – 503	Ascending or descending: 0 = ascending 1 = descending
504 – 505	Reserved for internal use

506 – 507	FOR expression pool length[1]
508 – 509	Reserved for internal use
510 – 511	Key expression pool length[1]
512 – 1023	Key expression pool (uncompiled)

[1] This information tracks the space used in the key expression pool.

Compact Index Interior Node Record

Byte offset	**Description**
00 – 01	Node attributes (any of the following numeric values or their sums): a. 0 – index node b. 1 – root node c. 2 – leaf node
02 – 03	Number of keys present (0, 1 or many)
04 – 07	Pointer to node directly to left of current node (on same level, -1 if not present)
08 – 11	Pointer to node directly to right of current node (on same level; -1 if not present)
12 – 511	Up to 500 characters containing the key value for the length of the key with a four-byte hexadecimal number (stored in normal left-to-right format): This node always contains the index key, record number and intra-index pointer.[2] The key/four-byte hexadecimal number combinations will occur the number of times indicated in bytes 02 – 03.

Compact Index Exterior Node Record

00 – 01	Node attributes (any of the following numeric values or their sums): 0 – index node 1 – root node 2 – leaf node

02 – 03	Number of keys present (0, 1 or many)
04 – 07	Pointer to the node directly to the left of current node (on same level; -1 if not present)
08 – 11	Pointer to the node directly to right of the current node (on same level; -1 if not present)
12 – 13	Available free space in node
14 – 17	Record number mask
18	Duplicate byte count mask
19	Trailing byte count mask
20	Number of bits used for record number
21	Number of bits used for duplicate count
22	Number of bits used for trail count
23	Number of bytes holding record number, duplicate count and trailing count
24 – 511	Index keys and information[2]

[2] Each entry consists of the record number, duplicate byte count and trailing byte count, all compacted. The key text is placed at the logical end of the node, working backwards, allowing for previous key entries.

See Also

Compound Index File Structure (.cdx) | Index File Structure (.idx) | Table File Structure (.dbc, .dbf, .frx, .lbx, .mnx, .pjx, .scx, .vcx) | Table Structures of Table Files (.dbc, .frx, .lbx, .mnx, .pjx, .scx, .vcx) | Memo File Structure (.FPT) | Macro File Format (.fky) | File Extensions and File Types