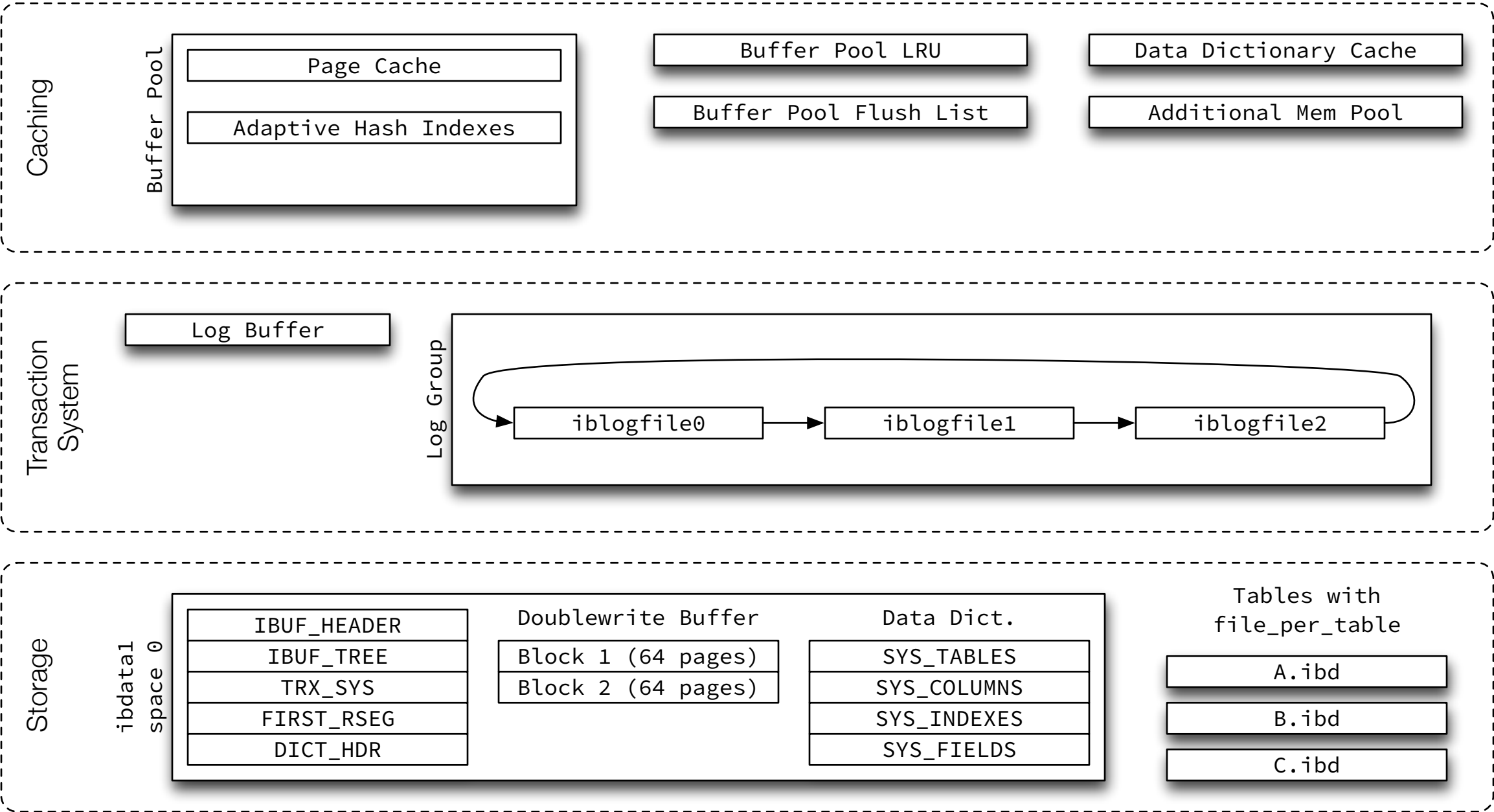
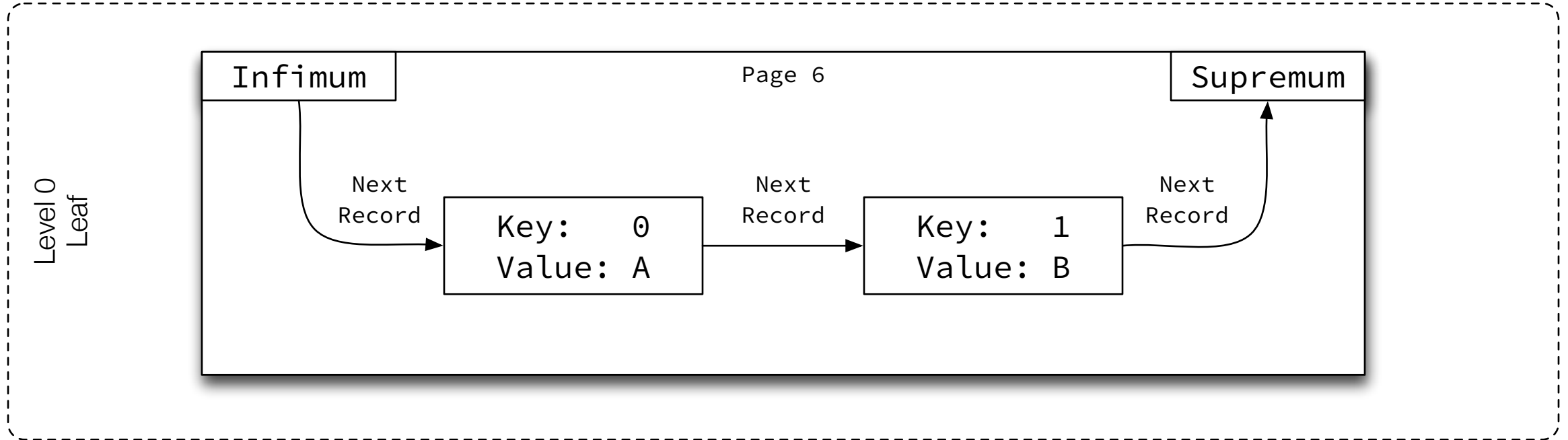


High-level Overview

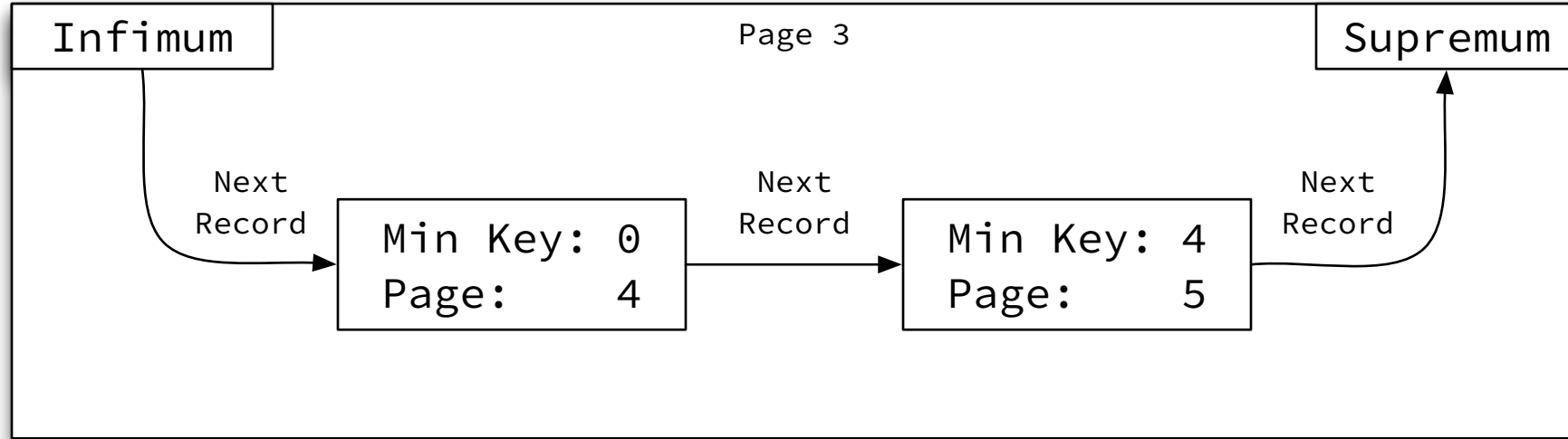


B+Tree Simplified Leaf Page

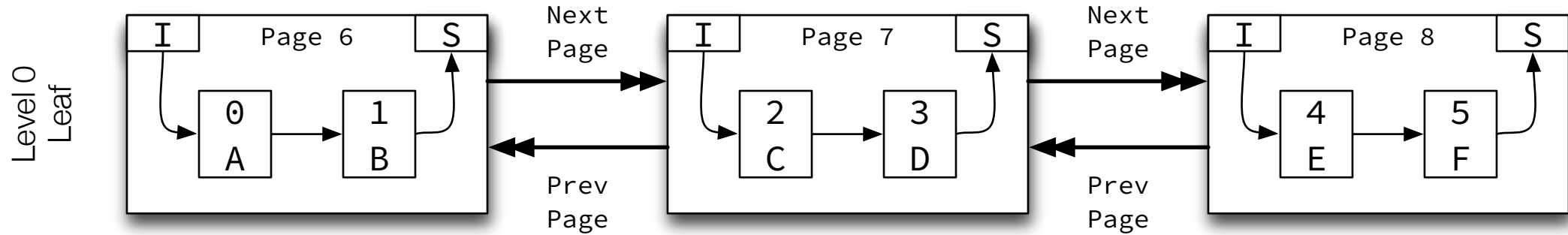


B+Tree Simplified Non-Leaf Page

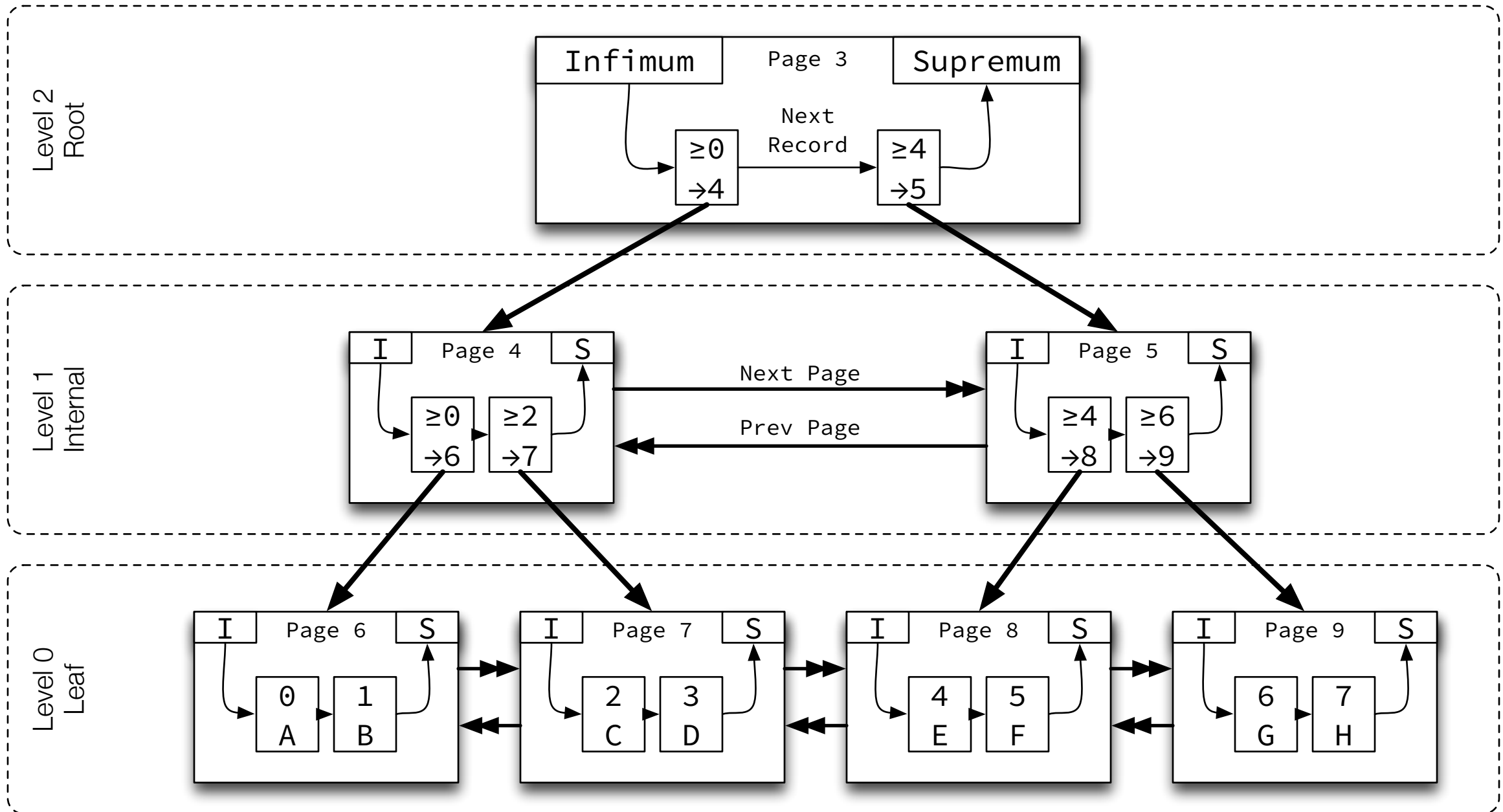
Level N
Non-Leaf



B+Tree Simplified Level



B+Tree Structure



Levels are numbered starting from 0 at the leaf pages, incrementing up the tree.

Pages on each level are doubly-linked with previous and next pointers in ascending order by key.

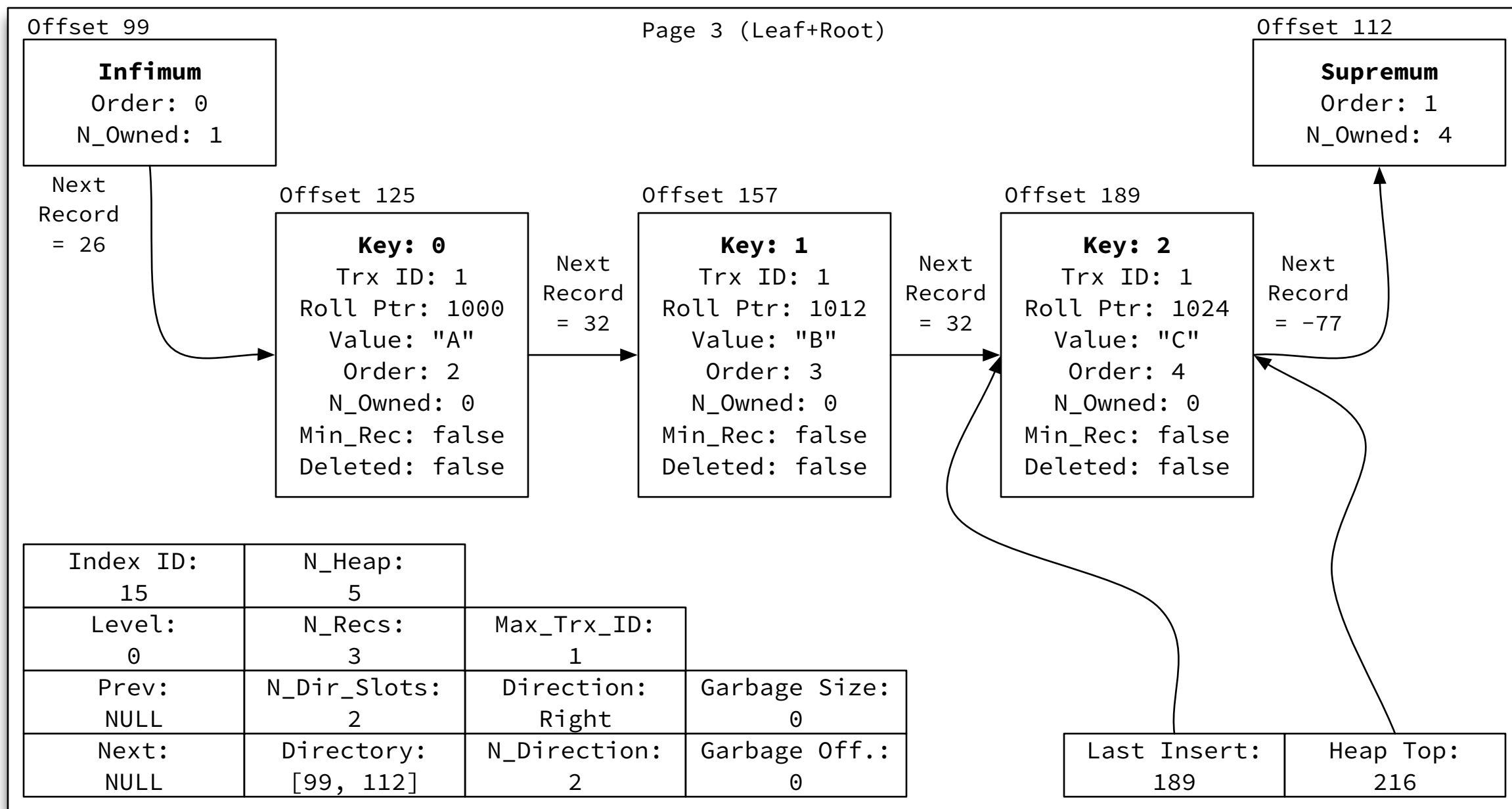
Records within a page are singly-linked with a next pointer in ascending order by key.

Infimum represents a value lower than any key on the page, and is always the first record in the singly-linked list of records.

Supremum represents a value higher than any key on the page, and is always the last record in the singly-linked list of records.

Non-leaf pages contain the minimum key of the child page and the child page number, called a "node pointer".

B+Tree Detailed Page Structure



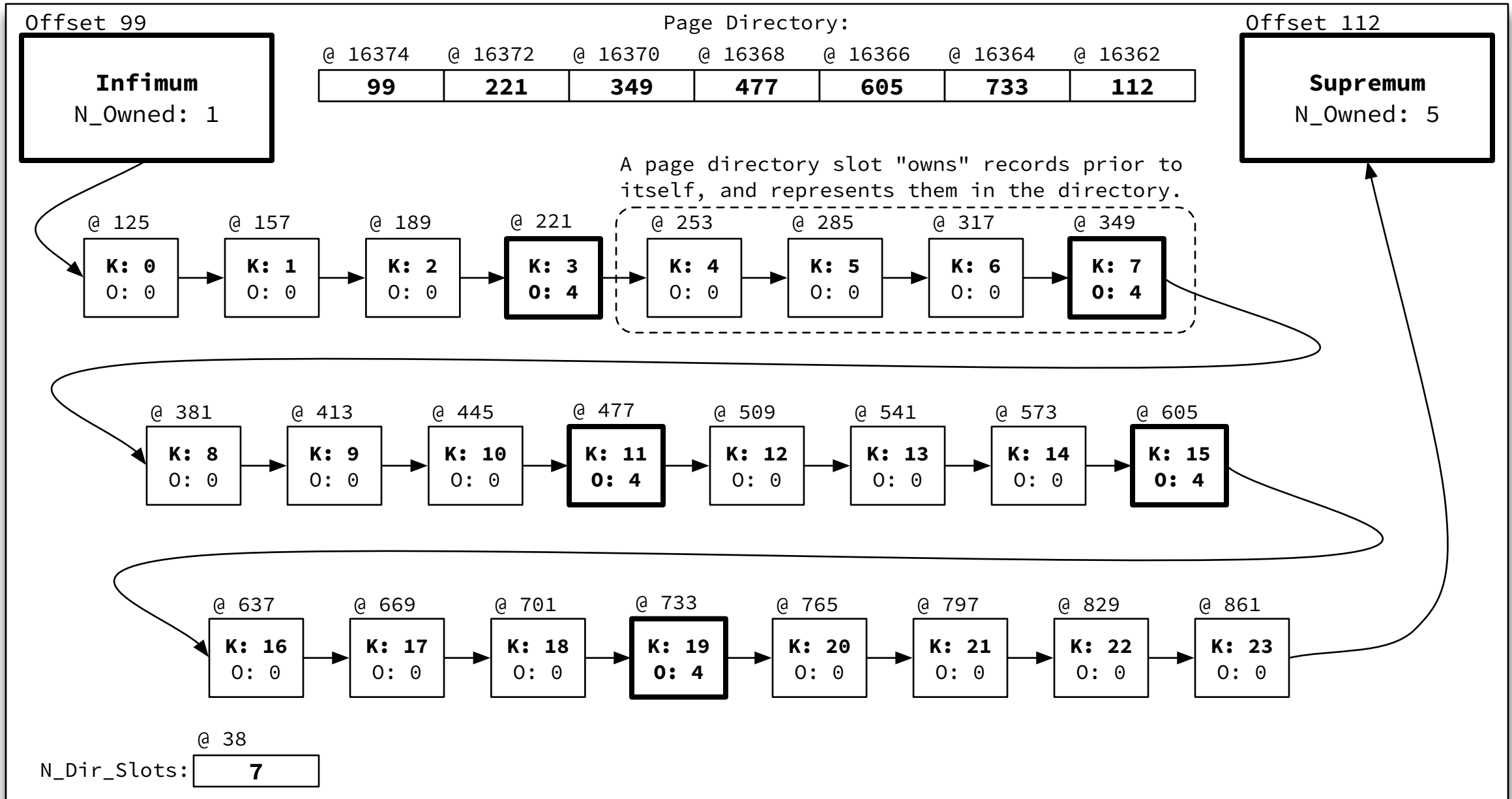
InnoDB table format is Barracuda with "compact" record structure, non-compressed.

Table created with: CREATE TABLE t (i INT NOT NULL, s CHAR(10) NOT NULL, PRIMARY KEY(i)) ENGINE=InnoDB;

Table populated with: INSERT INTO t (i, s) VALUES (0, "A"), (1, "B"), (2, "C");

Record size: 5 (header) + 4 (PK) + 6 (TRX_ID) + 7 (ROLL_PTR) + 10 (non-key fields) = 32 bytes

B+Tree Page Directory Structure



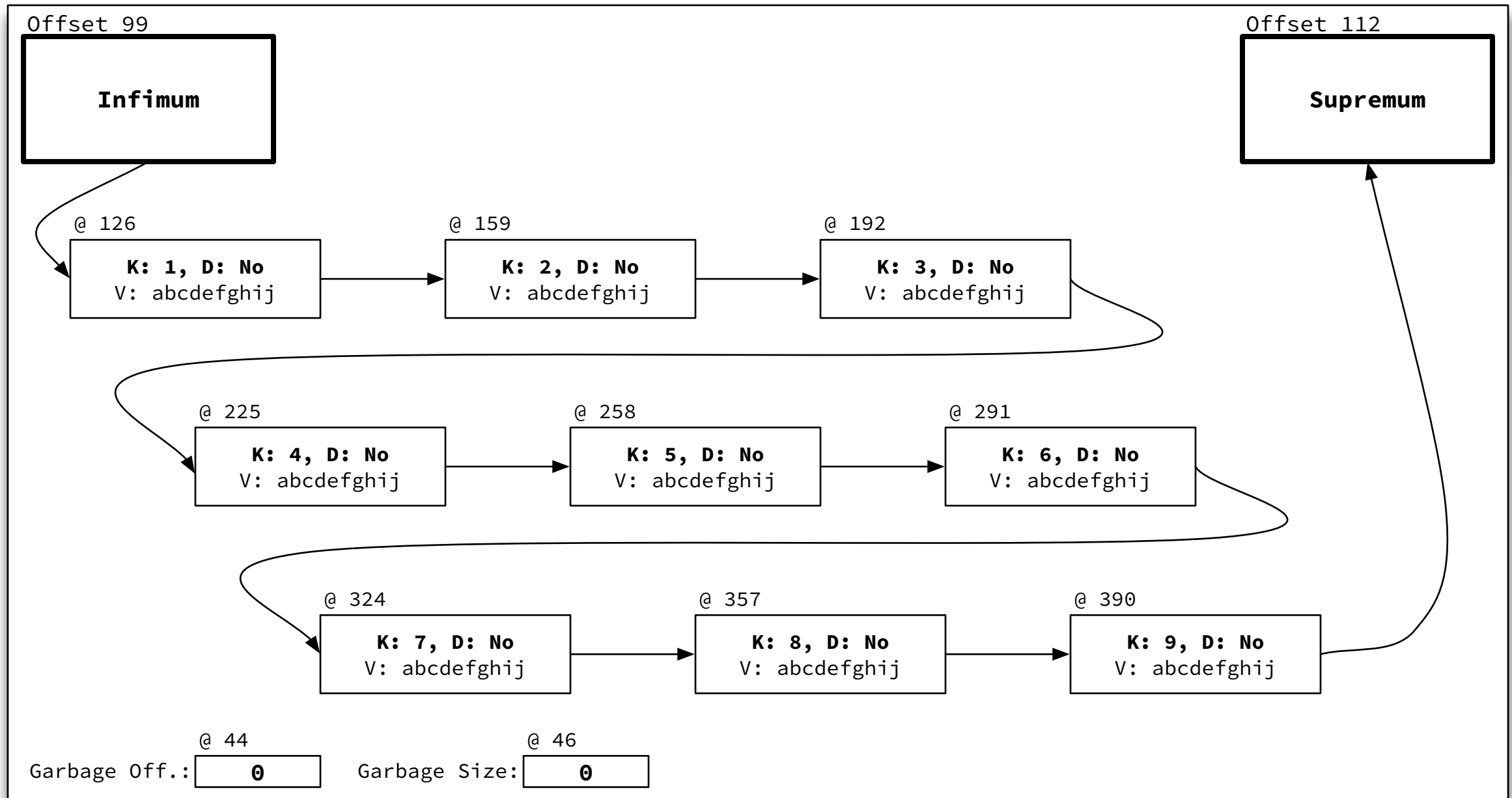
Infimum always owns only itself, so will always have a slot in the page directory with N_Owned = 1.

Supremum always owns the last few records in the page, and is allowed to own less than 4 records (if the page has fewer).

All directory slots will own a minimum of 4 and maximum of 8 records, except supremum, which may own fewer.

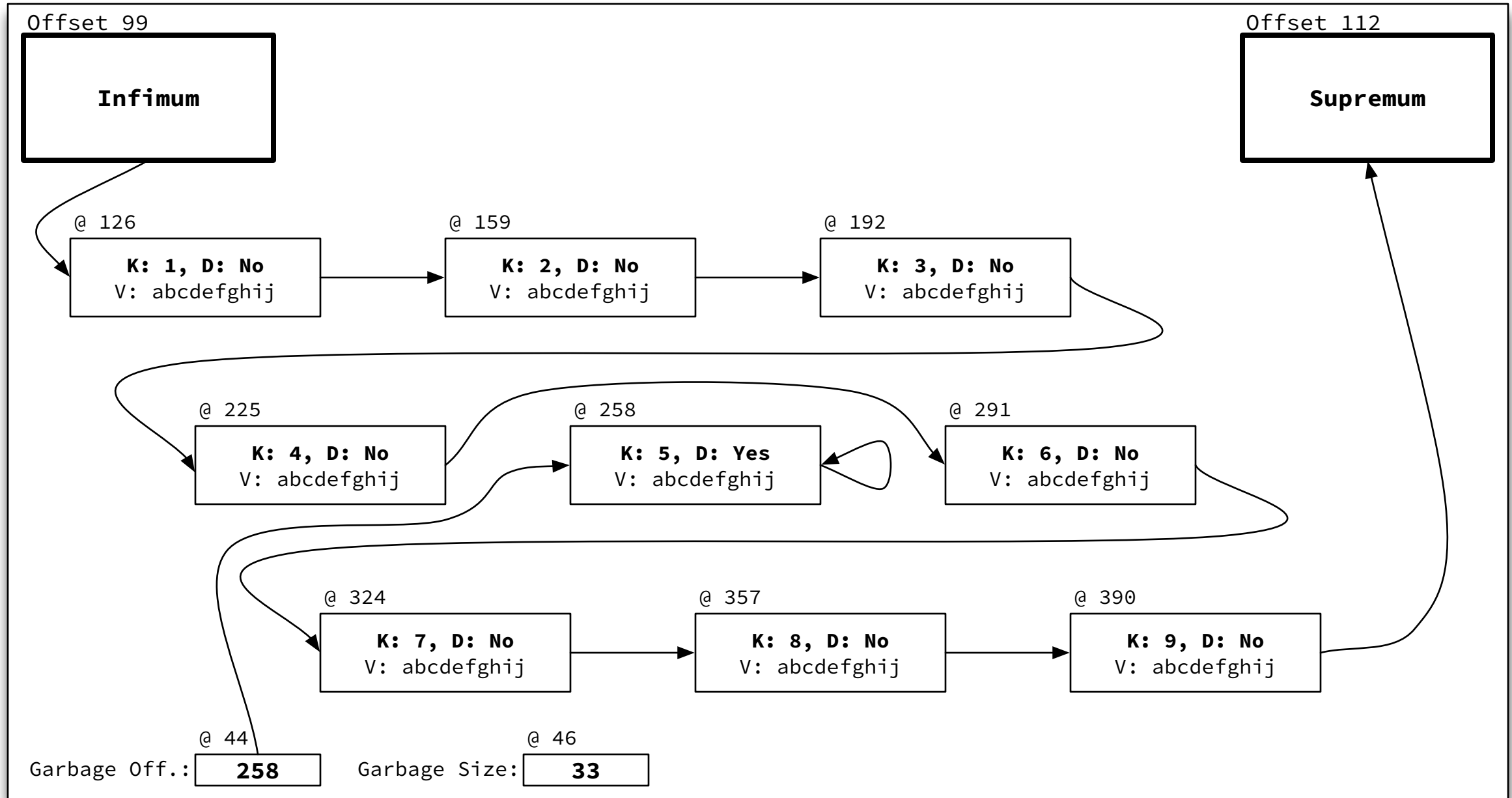
The page directory grows "downwards" from offset 16376, the beginning of the FIL trailer; the first directory entry starts at 16374.

B+Tree Record Initial State



SQL: create table t (i int not null, s varchar(100) not null, primary key(i)) engine=innodb;
SQL: insert into t (i, s) values (1, "abcdefghij"); for i in 1..9

B+Tree Record Delete 1



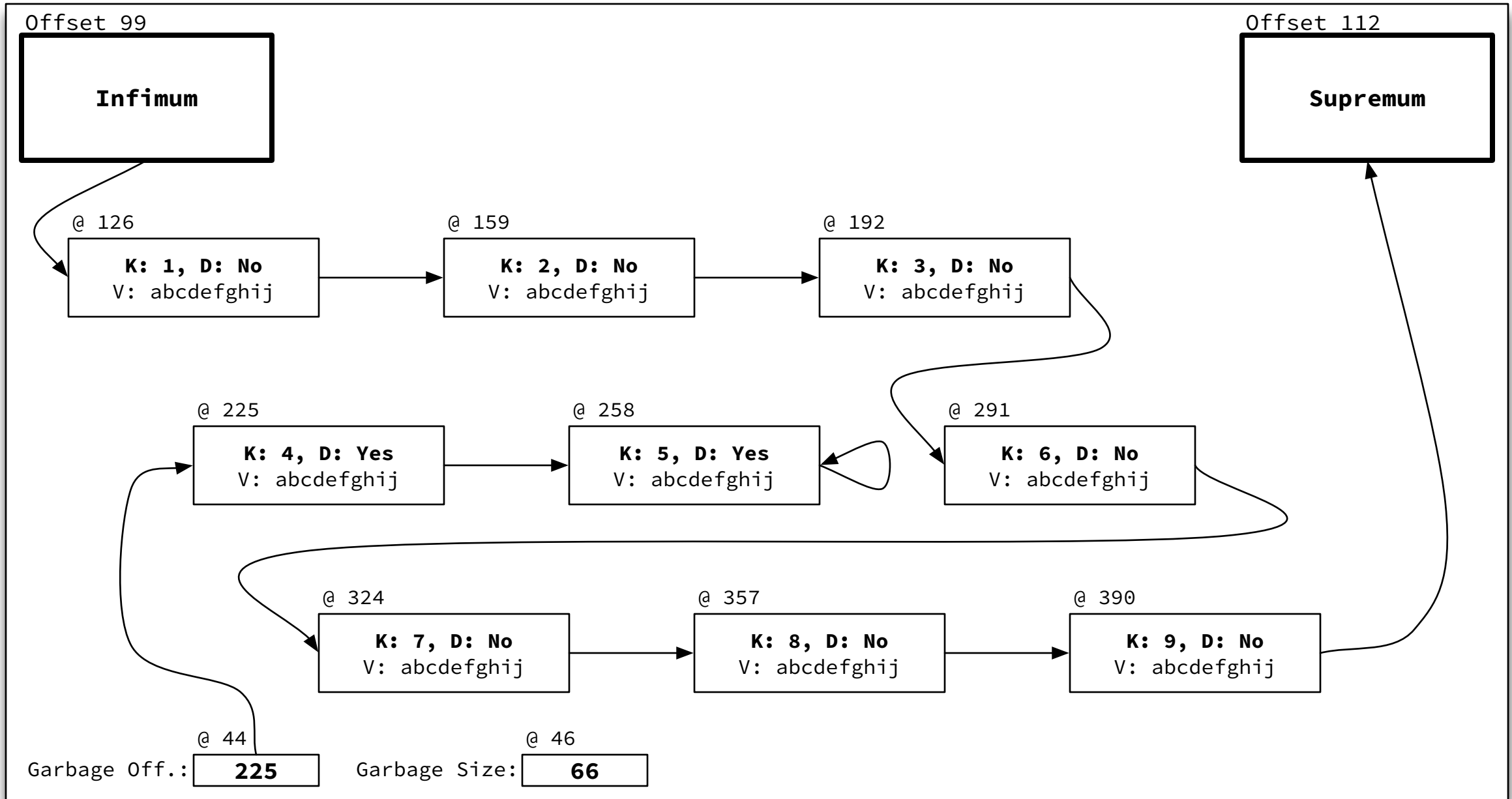
SQL: delete from t where i = 5;

Row is marked as deleted.

Garbage size is incremented by total row size.

Garbage offset is pointed to row, and row next pointer is pointed back to self.

B+Tree Record Delete 2



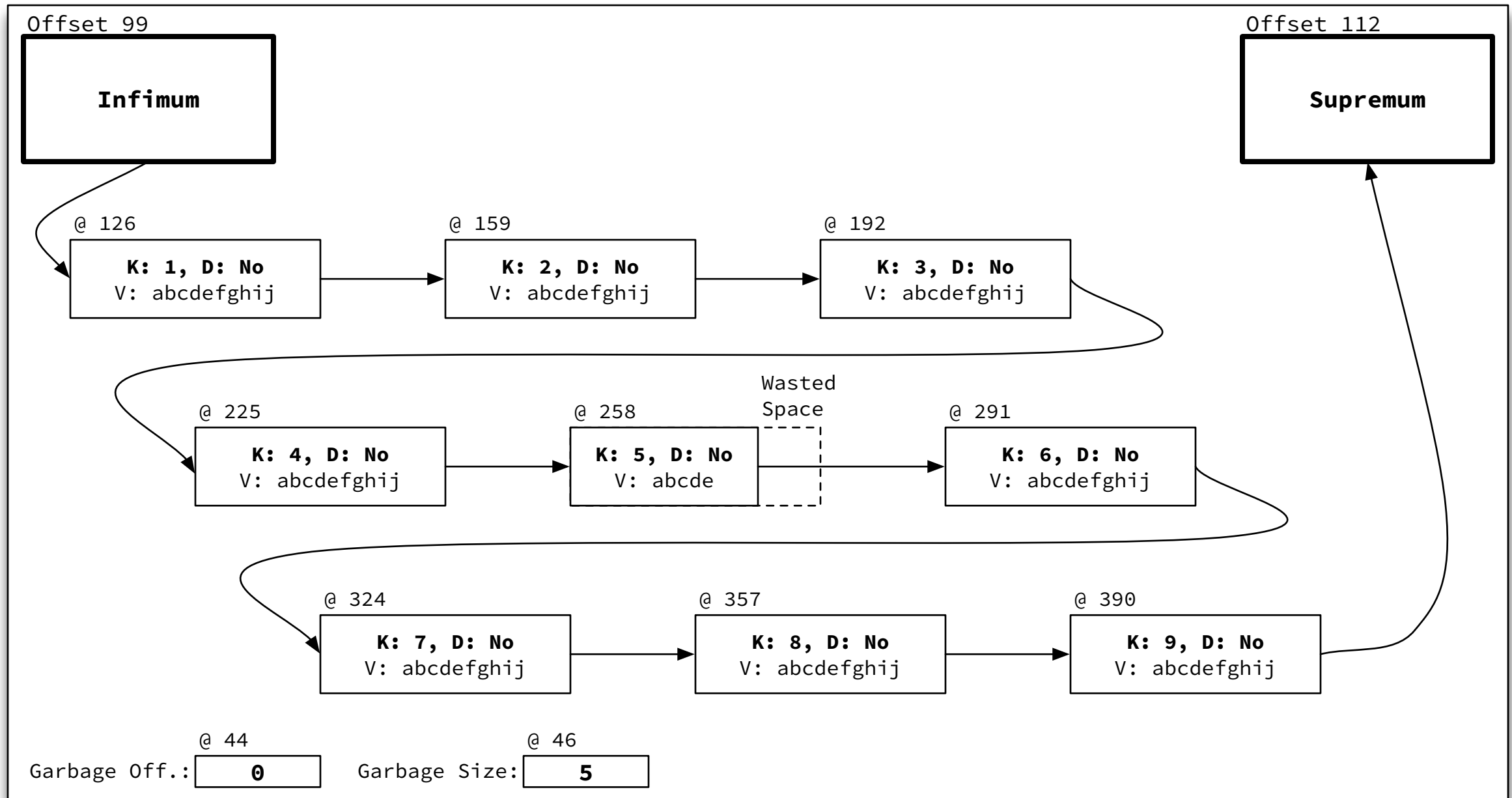
SQL: delete from t where i = 5; delete from t where i = 4;

Garbage size is incremented by total row size for each delete.

Garbage offset is pointed to row @ 258 initially, and row next pointer is pointed back to self.

Garbage offset is updated to row @ 225, and row next pointer is pointed to previous garbage offset (garbage is added to head of list).

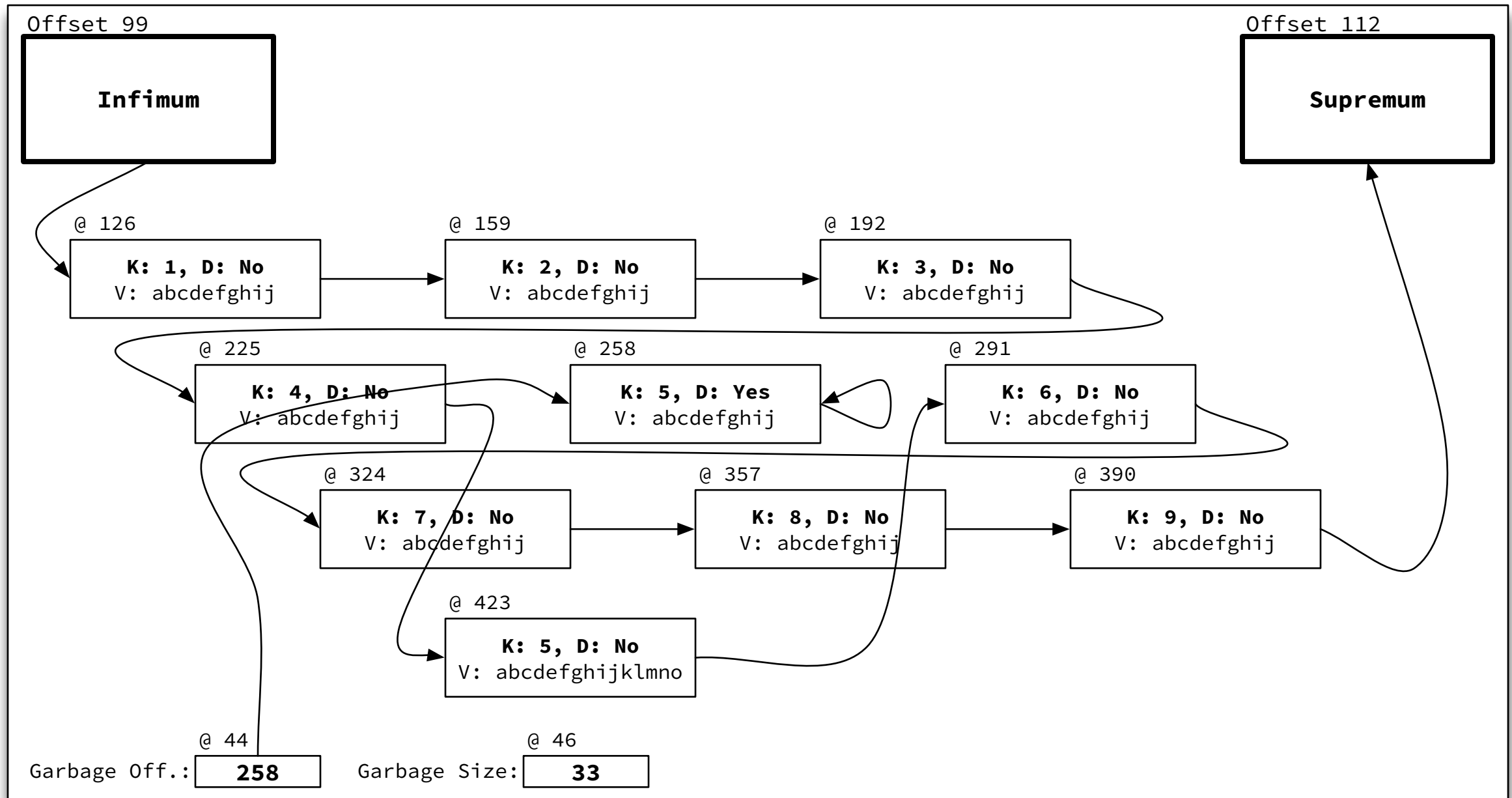
B+Tree Record Update - Smaller



SQL: update t set s="abcde" where i = 5;

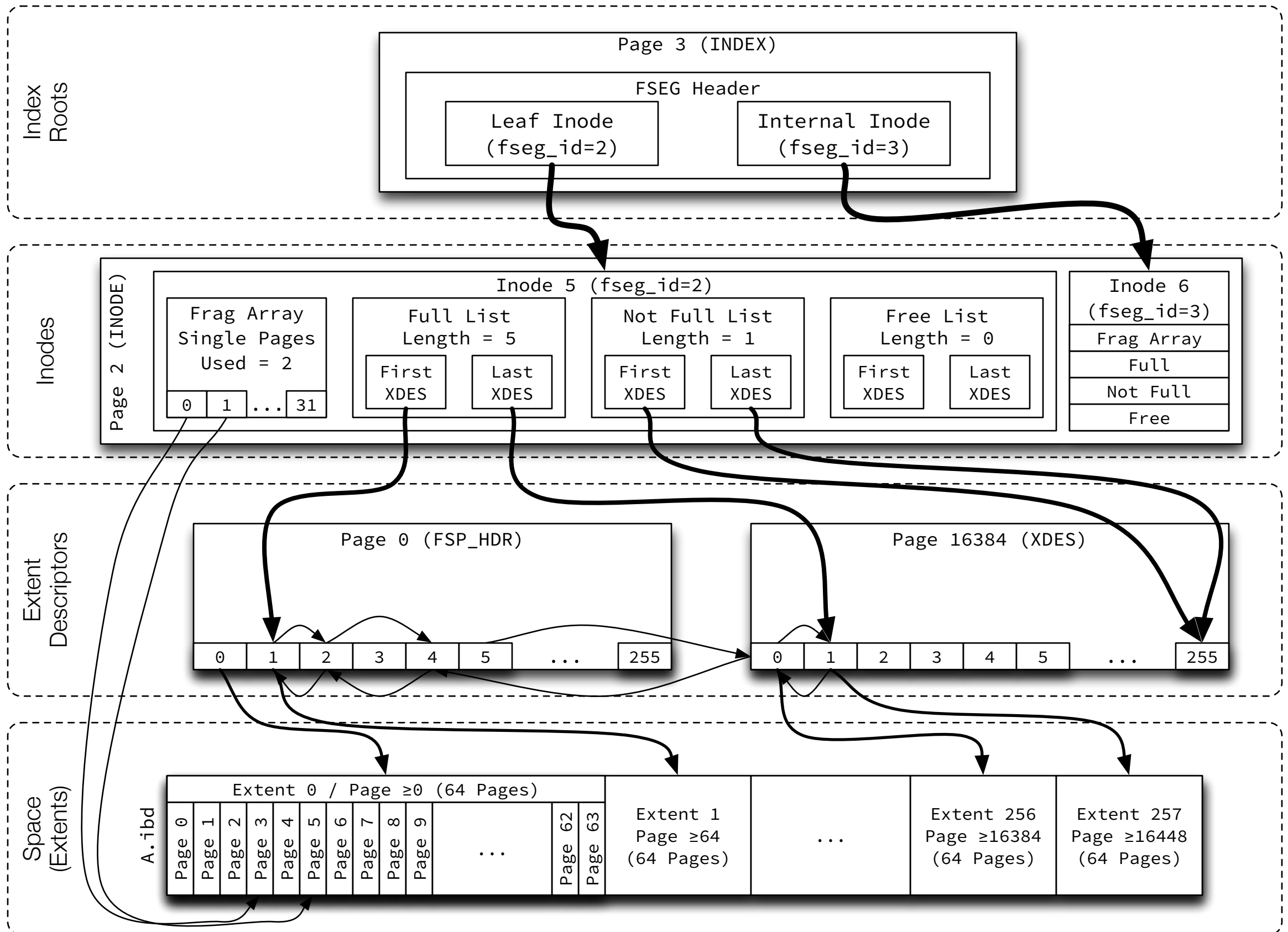
Garbage size is incremented by size of row shrinkage, but wasted space is not tracked in garbage list.

B+Tree Record Update - Larger

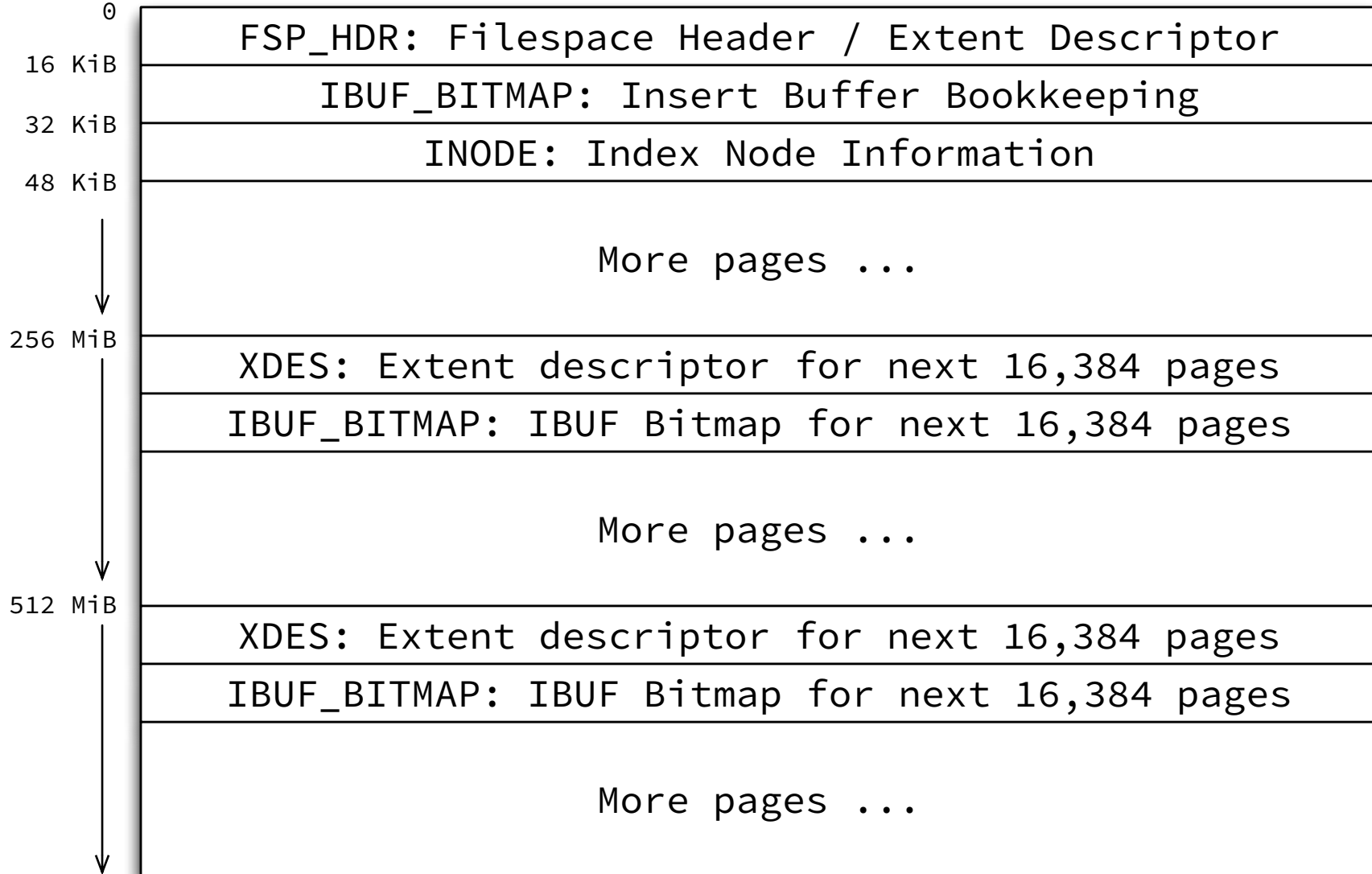


SQL: update t set s="abcdefghijklmno" where i = 5;
Row is deleted, and a new row is inserted into the heap.

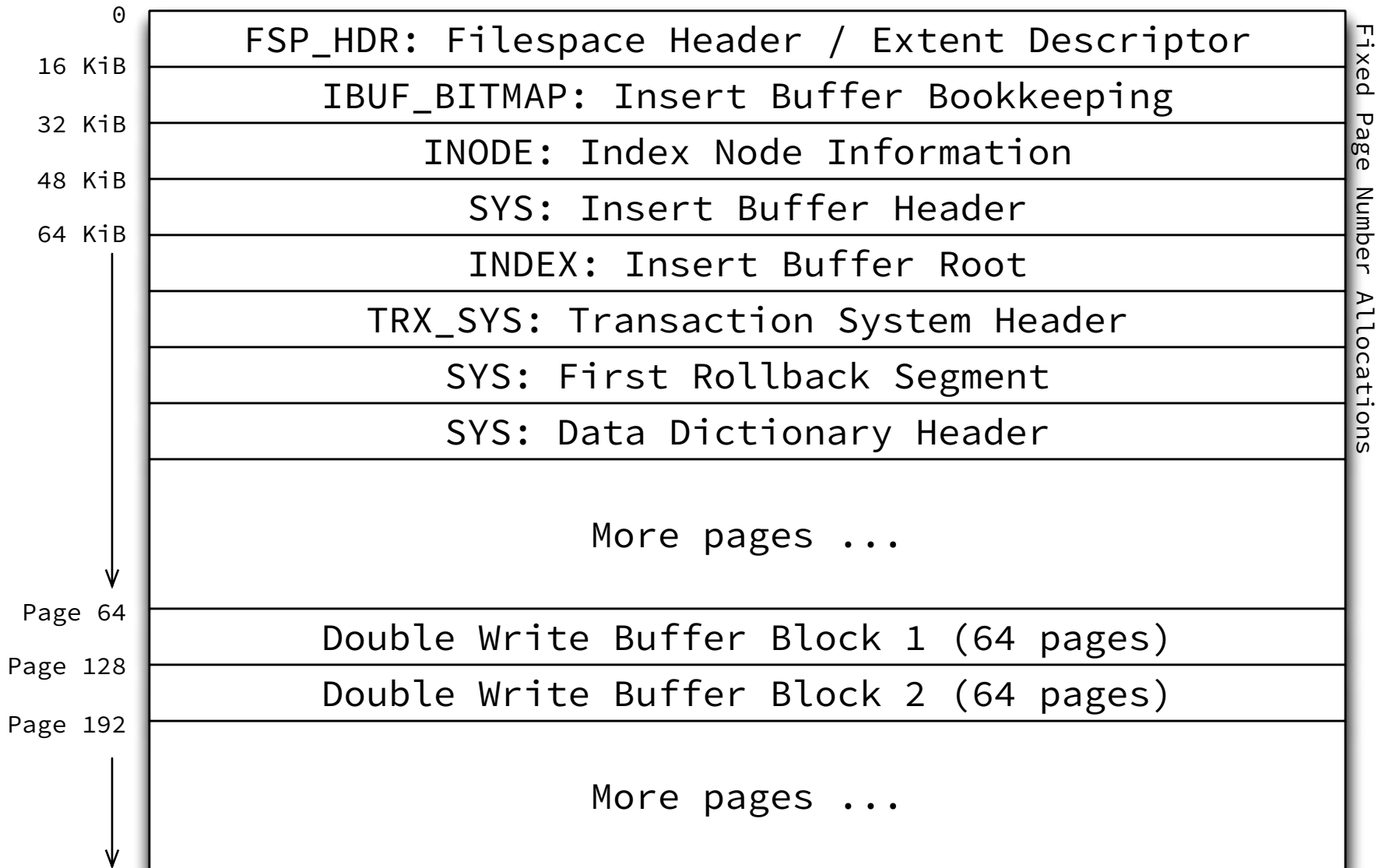
Index File Segment Structure



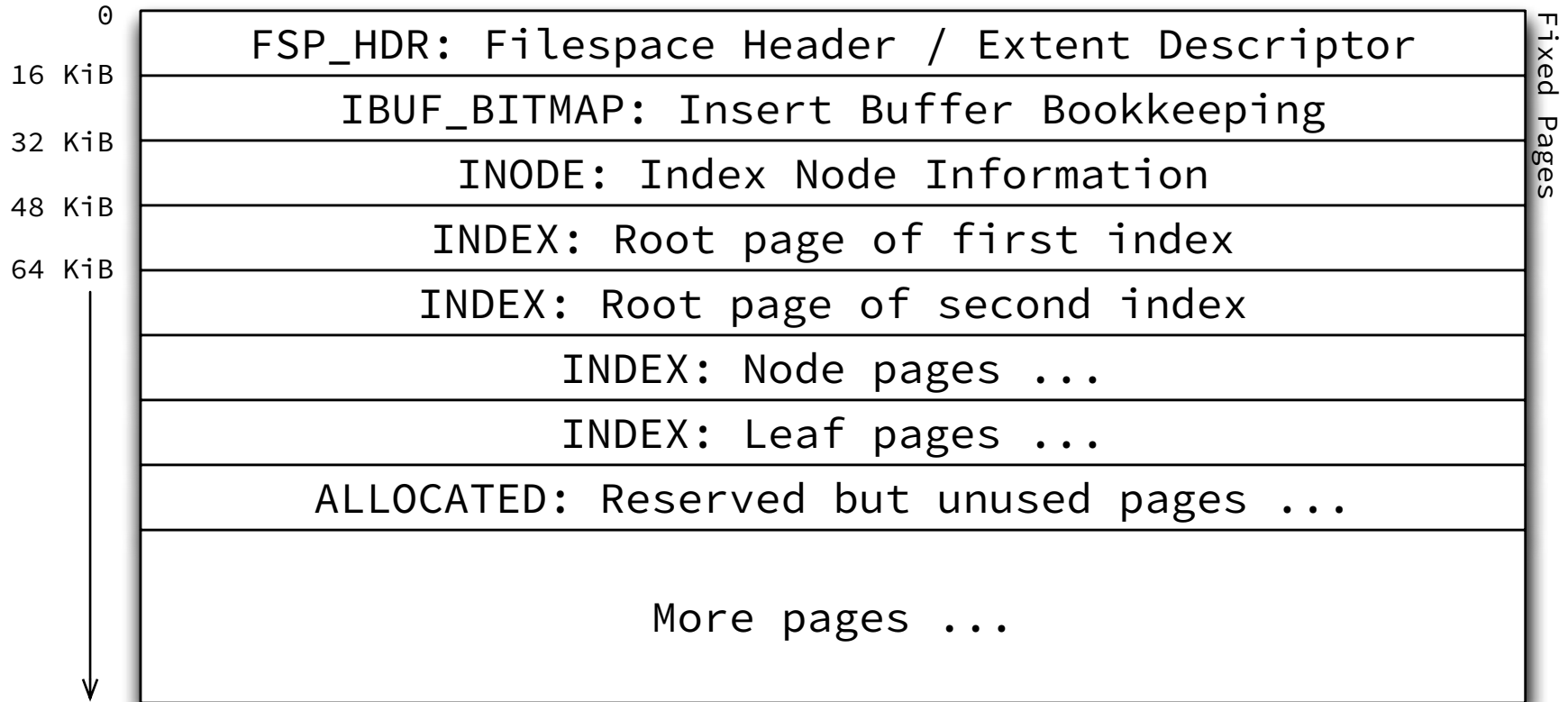
Space File Overview



ibdata1 File Overview



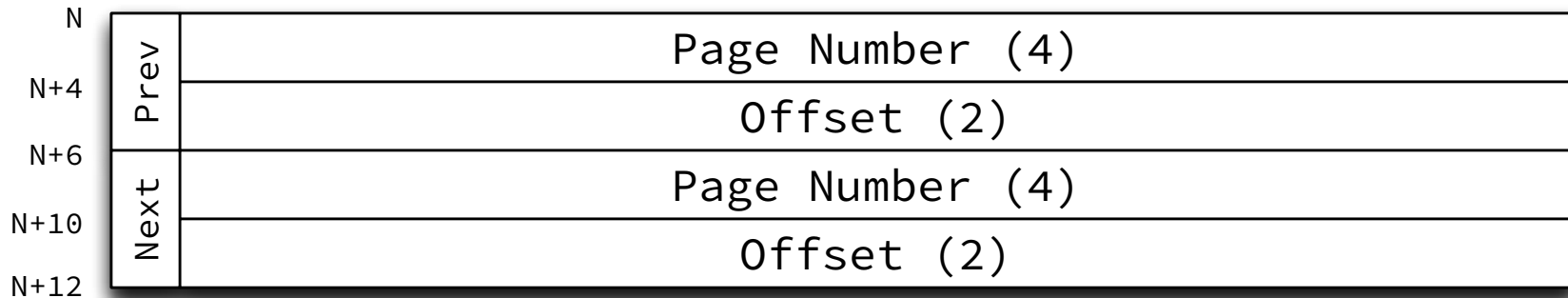
IBD File Overview



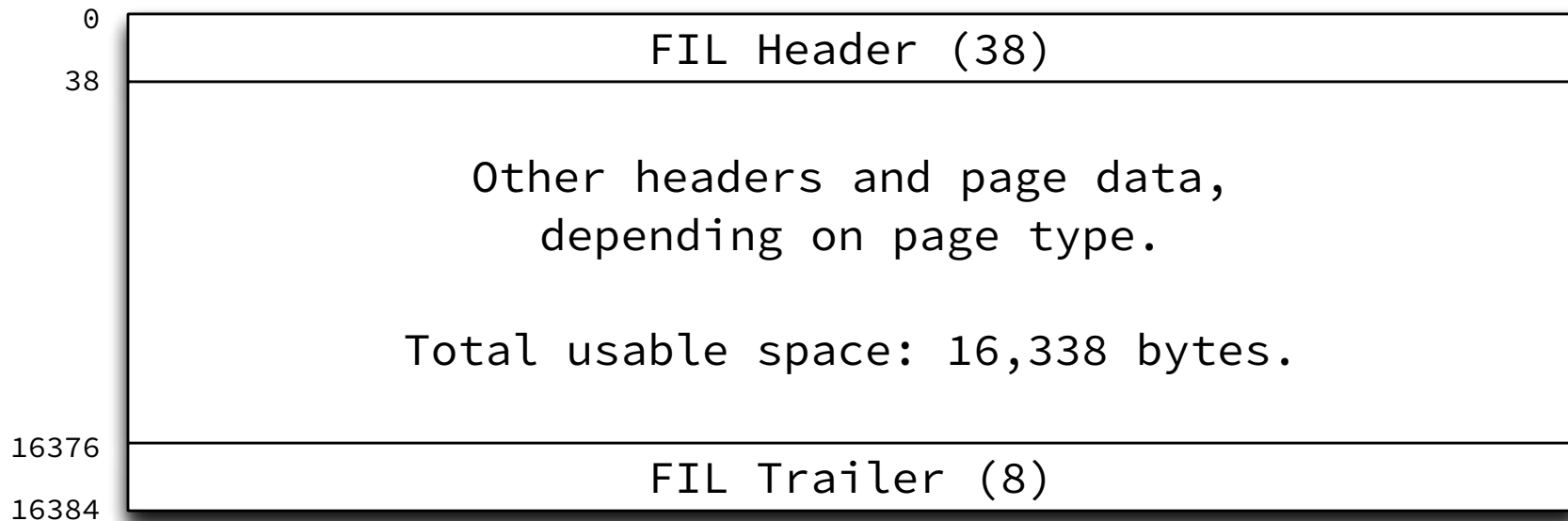
List Base Node

N	List Length (4)	
N+4	First	Page Number (4)
N+8		Offset (2)
N+10	Last	Page Number (4)
N+14		Offset (2)
N+16		

List Node



Basic Page Overview



FIL Header/Trailer

0	Checksum (4)
4	Offset (Page Number) (4)
8	Previous Page (4)
12	Next Page (4)
16	LSN for last page modification (8)
24	Page Type (2)
26	Flush LSN (0 except space 0 page 0) (8)
34	Space ID (4)
38	...
16376	Old-style Checksum (4)
16380	Low 32 bits of LSN (4)
16384	

FSP_HDR/XDES Overview

0	FIL Header (38)			
38	FSP Header (zero-filled for XDES pages) (112)			
150	XDES Entry	0 (pages	0– 63)	(40)
190	XDES Entry	1 (pages	64– 127)	(40)
230	XDES Entry	2 (pages	128– 191)	(40)
270	XDES Entry	3 (pages	192– 255)	(40)
310	...			
↓				
10310	XDES Entry	254 (pages	16256–16319)	(40)
10350	XDES Entry	255 (pages	16320–16383)	(40)
10390	(Empty Space: 5,986 bytes)			
16376	FIL Trailer (8)			
16384				

FSP Header

38	Space ID (4)
42	(Unused) (4)
46	Highest page number in file (size) (4)
50	Highest page number initialized (free limit) (4)
54	Flags (4)
58	Number of pages used in "FREE_FRAG" list (4)
62	List base node for "FREE" list (16)
78	List base node for "FREE_FRAG" list (16)
94	List base node for "FULL_FRAG" list (16)
110	Next Unused Segment ID (8)
118	List base node for "FULL_INODES" list (16)
134	List base node for "FREE_INODES" list (16)
150	

XDES Entry

N	File Segment ID (8)
N+8	List node for XDES list (12)
N+20	State (4)
N+24	Page State Bitmap (16) 2 bits per page, 1=free, 2=clean
N+40	

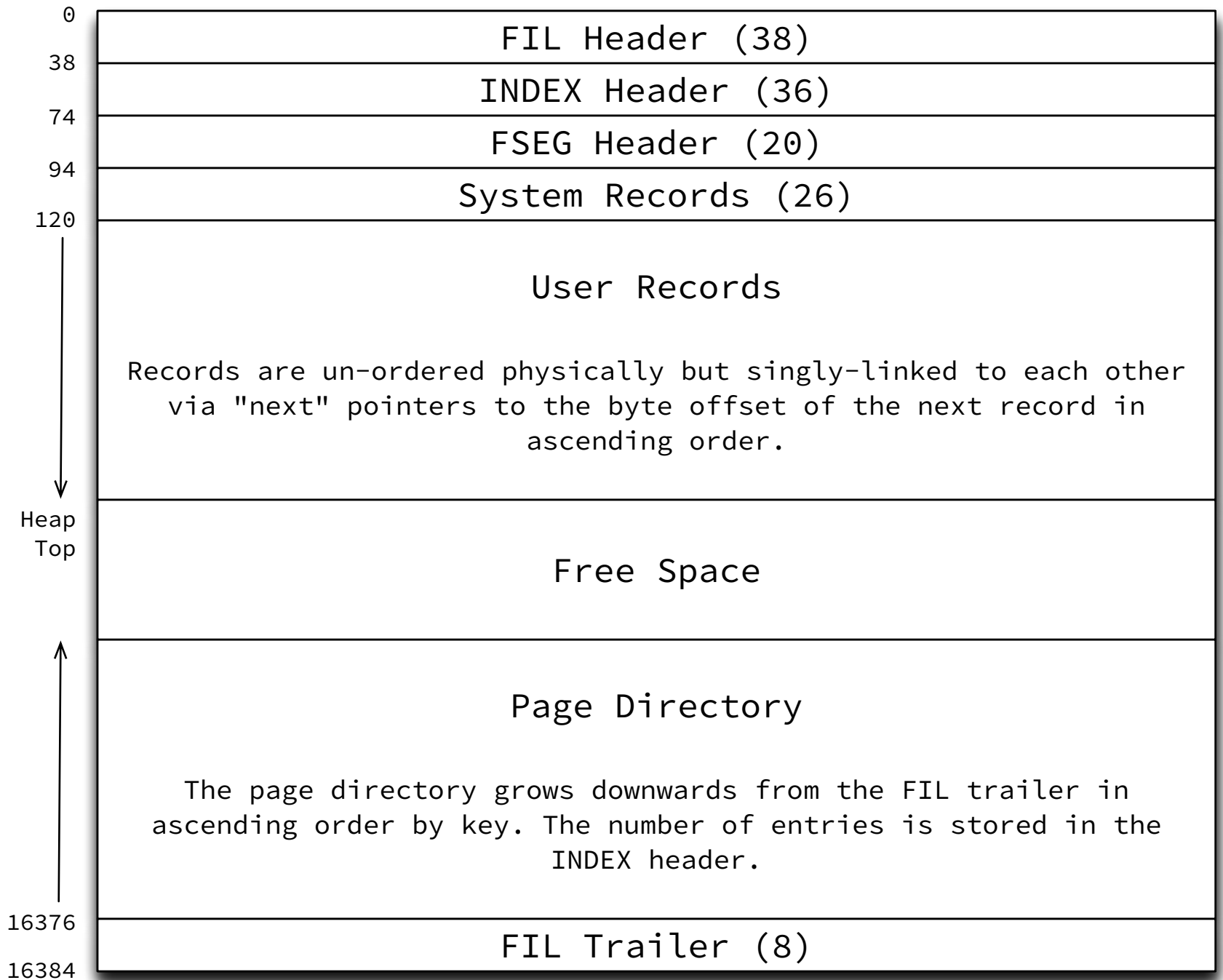
INODE Overview

0	FIL Header (38)
38	List node for INODE Page list (12)
50	INODE 0 (192)
242	INODE 1 (192)
434	INODE 2 (192)
626	...
↓	
15986	INODE 83 (192)
16178	INODE 84 (192)
16370	(Empty Space, 6 bytes)
16376	FIL Trailer (8)
16384	

INODE Entry

N	FSEG ID (8)
N+8	Number of used pages in "NOT_FULL" list (4)
N+12	List base node for "FREE" list (16)
N+28	List base node for "NOT_FULL" list (16)
N+44	List base node for "FULL" list (16)
N+60	Magic Number = 97937874 (4)
N+64	Fragment Array Entry 0 (4)
N+68	...
N+188	Fragment Array Entry 31 (4)
N+192	

INDEX Overview



INDEX Header

38	Number of Directory Slots (2)
40	Heap Top Position (2)
42	Number of Heap Records / Format Flag (2)
44	First Garbage Record Offset (2)
46	Garbage Space (2)
48	Last Insert Position (2)
50	Page Direction (2)
52	Number of Inserts in Page Direction (2)
54	Number of Records (2)
56	Maximum Transaction ID (8)
64	Page Level (2)
66	Index ID (4)
74	

FSEG Header

74	Leaf Pages Inode Space ID (4)
78	Leaf Pages Inode Page Number (4)
82	Leaf Pages Inode Offset (2)
84	Internal (non-leaf) Inode Space ID (4)
88	Internal (non-leaf) Inode Page Number (4)
92	Internal (non-leaf) Inode Offset (2)
94	

INDEX System Records

94	Info Flags (4 bits)
	Number of Records Owned (4 bits)
95	Order (13 bits)
	Record Type (3 bits)
97	Next Record Offset (2)
99	"infimum\0" (8)
107	Info Flags (4 bits)
	Number of Records Owned (4 bits)
108	Order (13 bits)
	Record Type (3 bits)
110	Next Record Offset (2)
112	"supremum" (8)
120	

INDEX Page Directory

$N - (d \times 2)$	Directory Slot d (2)
	...
$N - 4$	Directory Slot 1 (2)
$N - 2$	Directory Slot 0 (2)
N	

Record Format - Header

	Variable field lengths (1-2 bytes per var. field)
N-5	Nullable field bitmap (1 bit per nullable field)
	Info Flags (4 bits)
N-4	Number of Records Owned (4 bits)
	Order (13 bits)
N-2	Record Type (3 bits)
N	Next Record Offset (2)

Record Format - Clustered Key - Leaf Pages

N-5	Variable field lengths (1-2 bytes per var. field)
	Info Flags (4 bits)
N-4	Number of Records Owned (4 bits)
	Order (13 bits)
N-2	Record Type (3 bits)
N	Next Record Offset (2)
N+k	Cluster Key Fields (k)
N+k+6	Transaction ID (6)
N+k+13	Roll Pointer (7)
N+k+13+j	Non-Key Fields (j)

Record Format - Clustered Key - Non-Leaf Pages

	Variable field lengths (1-2 bytes per var. field)
N-5	Info Flags (4 bits)
	Number of Records Owned (4 bits)
N-4	Order (13 bits)
	Record Type (3 bits)
N-2	Next Record Offset (2)
N	Cluster Key Min. Key on Child Page (k)
N+k	Child Page Number (4)
N+k+4	

Record Format - Secondary Key - Leaf Pages

	Variable field lengths (1-2 bytes per var. field)
	Nullable field bitmap (1 bit per nullable field)
N-5	Info Flags (4 bits)
	Number of Records Owned (4 bits)
N-4	Order (13 bits)
	Record Type (3 bits)
N-2	Next Record Offset (2)
N	Secondary Key Fields (k)
N+k	Cluster Key Fields (j)
N+k+j	

Record Format - Secondary Key - Non-Leaf Pages

	Variable field lengths (1-2 bytes per var. field)
	Nullable field bitmap (1 bit per nullable field)
N-5	Info Flags (4 bits)
	Number of Records Owned (4 bits)
N-4	Order (13 bits)
	Record Type (3 bits)
N-2	Next Record Offset (2)
N	Secondary Key Min. Key on Child Page (k)
N+k	Cluster Key Fields (j)
N+k+j	Child Page Number (4)
N+k+j+4	