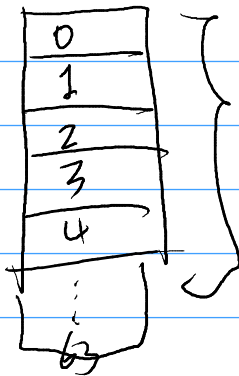


表空间: space\_id=(0), 页 page\_no=(1)

数据文件



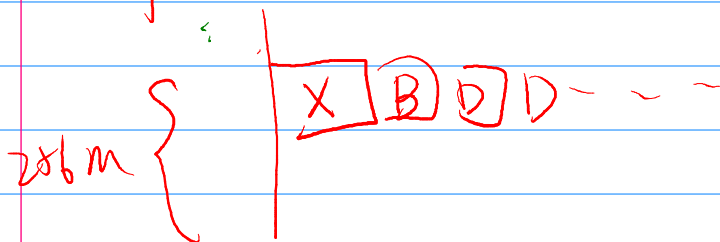
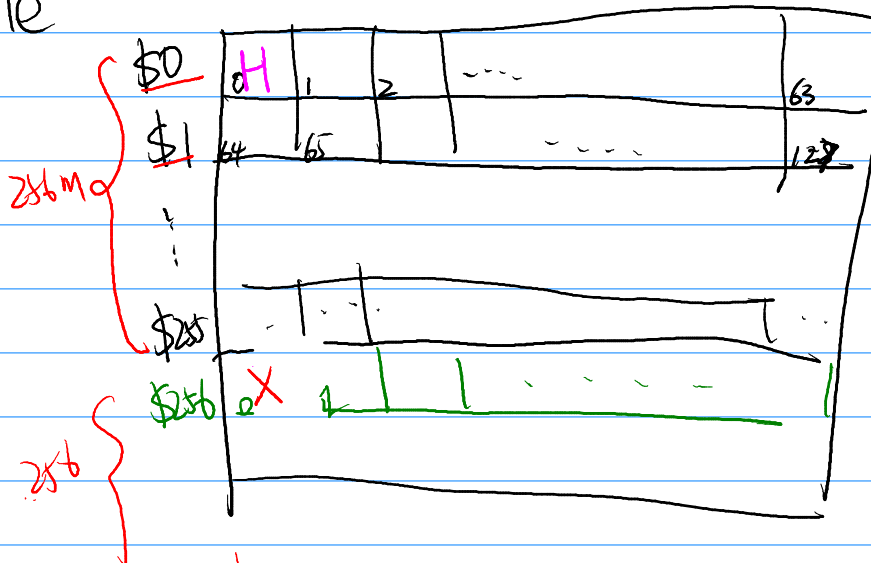
64页  $\Rightarrow$  1个区  $\Rightarrow$  256区

描述符

XDES FSRHDR

区  $\Rightarrow$  段 Segment

Datafile



```
mysql> select * from test111 limit 10;
```

<u>k1</u>	<u>c1</u>	<u>c2</u>
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10

10 rows in set (0.01 sec)

row format

Index

1000

1, 1, 1

row\_id

trx-id

roll\_ptr

```
mysql> show create table departments\G
```

\*\*\*\*\* 1. row \*\*\*\*\*

Table: departments

Create Table: CREATE TABLE `departments` (

dept\_no char(4) NOT NULL,

dept\_name varchar(40) NOT NULL,

PRIMARY KEY (`dept\_no`),

UNIQUE KEY `dept\_name` (`dept\_name`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

1 row in set (0.00 sec)

```
data/employees » ibr departments.ibd desc
```

COL1: name=dept\_no, type=STRING, nullable=false, data\_len=16, utf8\_def=char(4)

COL2: name=dept\_name, type=VARCHAR, nullable=false, data\_len=161, utf8\_def=varchar(40)

COL3: name=DB\_TRX\_ID, type=INT24, nullable=false, data\_len=6, utf8\_def=

COL4: name=DB\_ROLL\_PTR, type=LONGLONG, nullable=false, data\_len=7, utf8\_def=

IDX1: name=PRIMARY, type=IT\_PRIMARY, id=155, root=4, algorithm=IA\_BTREE

(1-1): column\_opx=0, col\_name=dept\_no, order=ORDER\_ASC, ele\_len=16, hidden=false, isnil=false, isv

(1-2): column\_opx=2, col\_name=DB\_TRX\_ID, order=ORDER\_ASC, ele\_len=-1, hidden=true, isnil=false, isv

(1-3): column\_opx=3, col\_name=DB\_ROLL\_PTR, order=ORDER\_ASC, ele\_len=-1, hidden=true, isnil=false, isv

(1-4): column\_opx=1, col\_name=dept\_name, order=ORDER\_ASC, ele\_len=-1, hidden=true, isnil=false, isv

IDX2: name=dept\_name, type=IT\_UNIQUE, id=156, root=5, algorithm=IA\_BTREE

(2-1): column\_opx=1, col\_name=dept\_name, order=ORDER\_ASC, ele\_len=160, hidden=false, isnil=false, isv

(2-2): column\_opx=0, col\_name=dept\_no, order=ORDER\_ASC, ele\_len=-1, hidden=true, isnil=false, isv

data/employees »

SID1

```

"indexes": [
  {
    "name": "PRIMARY",
    "hidden": false,
    "is_generated": false,
    "ordinal_position": 1,
    "comment": "",
    "options": "flags=0;",
    "se_private_data": "id=155;root=4;space_id=3;table_id=1065;trx_id=1300;",
    "type": 1,
    "algorithm": 2,
    "is_algorithm_explicit": false,
    "is_visible": true,
  }
]

```

Storage Engine

根页级 page no

departments

叶子

非叶子

2 段

```

fseg_hdr: FSegHeader {
  addr: 0x004a@(74),
  leaf_space_id: 3,
  leaf_page_no: 2,
  leaf_offset: 626,
  nonleaf_space_id: 3,
  nonleaf_page_no: 2,
  nonleaf_offset: 434,
}

```

```

INodeEntry {
  addr: 0x0272@(626),
  inode_seq: 3,
  fseg_id: 4,
  fseg_not_full_n_used: 0,
  fseg_free: FlstBaseNode {
    addr: 0x027e@(638),
    len: 0,
    first: FilAddr { addr: 0x0282@(642), page: },
    last: FilAddr { addr: 0x0288@(648), page: },
  },
  fseg_not_full: FlstBaseNode {
    addr: 0x028e@(654),
    len: 0,
    first: FilAddr { addr: 0x0292@(658), page: },
    last: FilAddr { addr: 0x0298@(664), page: },
  },
  fseg_full: FlstBaseNode {
    addr: 0x029e@(670),
    len: 0,
    first: FilAddr { addr: 0x02a2@(674), page: },
    last: FilAddr { addr: 0x02a8@(680), page: },
  },
  fseg_magic_n: 97937874,
  fseg_frag_arr: [],
}

```

```

INodeEntry {
  addr: 0x01b2@(434),
  inode_seq: 2,
  fseg_id: 3,
  fseg_not_full_n_used: 0,
  fseg_free: FlstBaseNode {
    addr: 0x01be@(446),
    len: 0,
    first: FilAddr { addr: 0x01c2@(450), page: None, boffset: 0 },
    last: FilAddr { addr: 0x01c8@(456), page: None, boffset: 0 },
  },
  fseg_not_full: FlstBaseNode {
    addr: 0x01ce@(462),
    len: 0,
    first: FilAddr { addr: 0x01d2@(466), page: None, boffset: 0 },
    last: FilAddr { addr: 0x01d8@(472), page: None, boffset: 0 },
  },
  fseg_full: FlstBaseNode {
    addr: 0x01de@(478),
    len: 0,
    first: FilAddr { addr: 0x01e2@(482), page: None, boffset: 0 },
    last: FilAddr { addr: 0x01e8@(488), page: None, boffset: 0 },
  },
  fseg_magic_n: 97937874,
  fseg_frag_arr: [4],
}

```

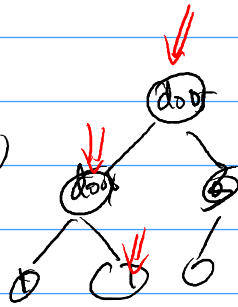
碎片区

B+ 树 (索引)

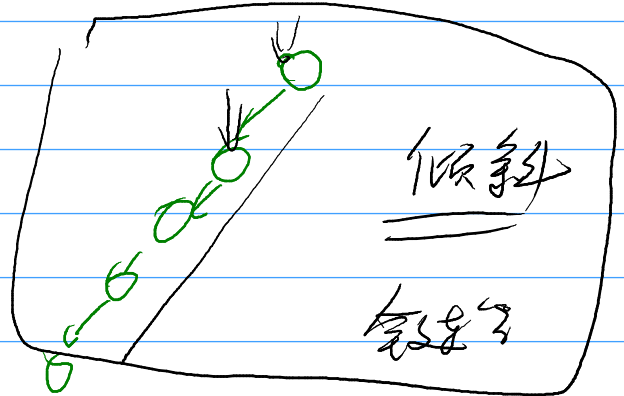
$\swarrow$   
[do01, do02, ..., do09]

搜  $O(n)$

树 (二叉树)

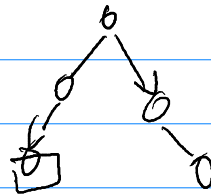


$O(\log n)$

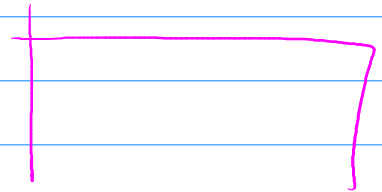


AVL 平衡树

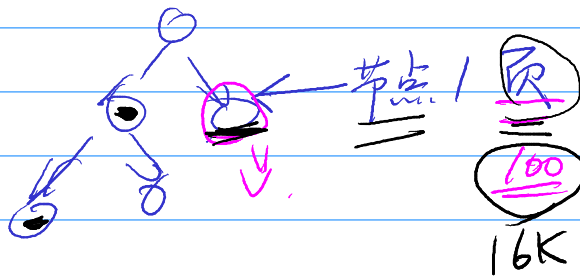
树  $\Downarrow$



红黑树 rb tree



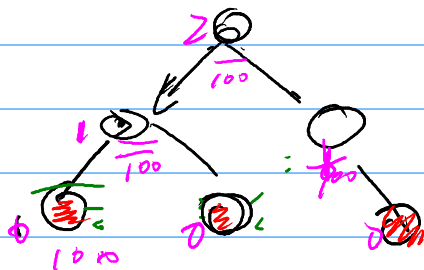
树  
B 树



磁盘 I/O  
块读写

100  
16K

B+ 树



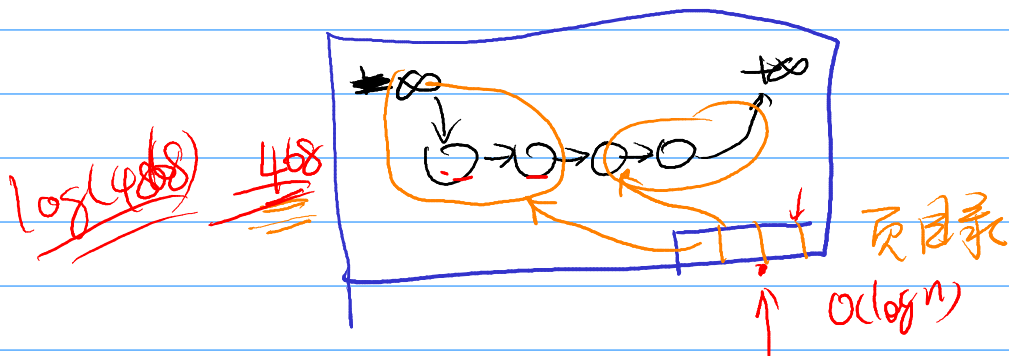
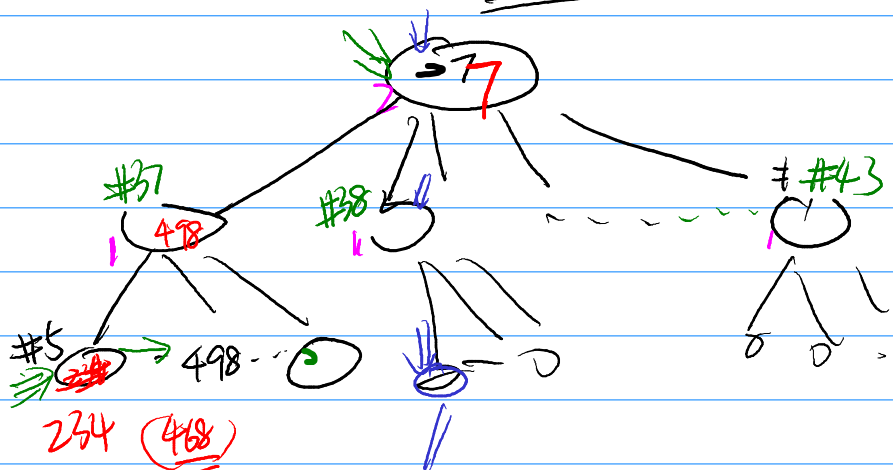
1000 x 100 x 100

Index page level

data/employees » ibr salaries.ibd dump

#0	[FSP_HD]	[IBUF_B]	[INODE]	[SDI]	[2]	[7]	[0, 234]	[0, 468]	[0, 468]
#8	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#16	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#24	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#32	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[1, 492]	[1, 984]	[1, 984]	[1, 984]
#40	[1, 984]	[1, 984]	[1, 984]	[1, 666]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]
#48	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]
#56	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]	[ALLOCA]
#64	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#72	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#80	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#88	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#96	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#104	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#112	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#120	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#128	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#136	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]
#144	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]	[0, 468]

INDEX B+



B+ 树 (有序)

Key: —

范围查询 查询

非叶子

Hash 索引 无序

Key 无序, OLAP

叶子  
非叶子

```

from_date => Date(1994-06-24)
DB_TRX_ID => TrxId(0x0000000005d9)
DB_ROLL_PTR => RbPtr(RollPtr { rptr: 0x82000000c40198, insert: true, seg_id: 2, page_no: 196, boffset: 408 })
salary => I32(75994)
to_date => Date(1995-06-24)
***** Row 10 *****
emp_no => I32(10001)
from_date => Date(1995-06-24)
DB_TRX_ID => TrxId(0x0000000005d9)
DB_ROLL_PTR => RbPtr(RollPtr { rptr: 0x82000000c401a9, insert: true, seg_id: 2, page_no: 196, boffset: 425 })
salary => I32(76884)
to_date => Date(1996-06-23)
ONLY dump 10 of 234 rows, use '--limit num' to dump more
data/employees » ibr salaries ibd dump 4
***** Row 1 *****
emp_no => I32(10001)
from_date => Date(1986-06-26)
NODE_PTR => PageNo(37)
***** Row 2 *****
emp_no => I32(34219)
from_date => Date(1994-05-09)
NODE_PTR => PageNo(38)
***** Row 3 *****
emp_no => I32(82707)
from_date => Date(1994-09-22)
NODE_PTR => PageNo(39)

```

Select \* from Salaries where emp\_no='0' and from\_date > '2000-01-01'

回表: ① 非叶子查到 key

② key 返回表主键索引再查一次

