数据库性能优化之索引

（上半部分：关于索引、单列索引、区分度）

关于我

- 王子亭
- Node.js 开发者
- LeanCloud
- https://jysperm.me
- GitHub: jysperm
- Decentralization

岗位要求：

- 精通 LNMP 编程，有 5 年以上项目经验；
- 了解数据库原理，能够进行索引和查询的性能调优；
- 熟悉 Linux 操作系统及常用 Shell 命令，能够在 Linux 下进行故障排查以及一些数据分析；

查询

Microsoft Date	Microsoft Open	Microsoft High	Microsoft Low	Microsoft Close	Microsoft Volume	Google Date	Google Open	Google High	Google Low	Google Close	Google Volume	Amazon Date	Amazon Open	Amazon High	Amazon Low	Amazon Close	Amazon Volume	Apple Date	Apple Open	Ap Hig
31-Dec-14	46.73	47.44	46.45	46.45	21552450	31-Dec-14	537.74	538.4	530.2	530.66	1236445	31-Dec-14	311.55	312.98	310.01	310.35	2057766	31-Dec-14	112.82	11
30-Dec-14	47.44	47.62	46.84	47.02	16384692	30-Dec-14	534.96	537.84	533.61	535.28	1048631	30-Dec-14	309.91	313.94	309.34	310.3	2093023	30-Dec-14	113.64	11
29-Dec-14	47.7	47.78	47.26	47.45	14439518	29-Dec-14	540.5	543.93	537.16	537.31	2218632	29-Dec-14	307.85	314.27	306.58	312.04	3009046	29-Dec-14	113.79	11
26-Dec-14	48.41	48.41	47.82	47.88	13197817	26-Dec-14	536.93	543.25	535.49	541.52	1113464	26-Dec-14	305	310.78	303.81	309.09	2893801	26-Dec-14	112.1	11
24-Dec-14	48.64	48.64	48.08	48.14	11442790	24-Dec-14	538.82	540.29	535.1	536.93	737848	24-Dec-14	306.38	307	302.88	303.03	1518107	24-Dec-14	112.58	11
23-Dec-14	48.37	48.8	48.13	48.45	23656529	23-Dec-14	534.51	542.3	533.72	538.77	2877222	23-Dec-14	306.98	307.49	303.25	306.28	2718359	23-Dec-14	113.23	11
22-Dec-14	47.78	48.12	47.71	47.98	26565984	22-Dec-14	520.61	532.97	520.59	532.3	3319461	22-Dec-14	301.94	307.36	301.94	306.54	4003827	22-Dec-14	112.16	11
19-Dec-14	47.63	48.1	47.17	47.66	64551182	19-Dec-14	516.99	520.81	508.86	520.04	4872059	19-Dec-14	296.91	301.54	295.52	299.9	8709129	19-Dec-14	112.26	11
18-Dec-14	46.58	47.52	46.34	47.52	40105550	18-Dec-14	515.99	516.25	506.56	514.62	3876732	18-Dec-14	304.01	304.5	293.25	297.73	7738067	18-Dec-14	111.87	11
17-Dec-14	45.05	45.94	44.9	45.74	34970865	17-Dec-14	499.86	509.07	499.11	506.45	3639437	17-Dec-14	296.37	299.67	293.03	298.88	4433505	17-Dec-14	107.12	10
16-Dec-14	45.9	46.34	45.13	45.16	47801392	16-Dec-14	513.6	515.58	497.19	498.16	4349882	16-Dec-14	304.35	304.49	295.01	295.06	6501252	16-Dec-14	106.37	11
15-Dec-14	47.2	47.67	46.55	46.67	29247761	15-Dec-14	523.76	525.37	515.4	515.84	2621977	15-Dec-14	308.87	310.86	302.15	306.07	3841577	15-Dec-14	110.7	1
12-Dec-14	46.78	47.73	46.67	46.95	34248371	12-Dec-14	527.44	532.24	521.46	521.51	2371543	12-Dec-14	303.99	310.64	303.01	307.32	3162322	12-Dec-14	110.46	11
11-Dec-14	47.08	47.74	46.68	47.17	29061918	11-Dec-14	530.01	537.04	529.16	532.11	2073290	11-Dec-14	307.89	312.64	306.01	307.36	3272919	11-Dec-14	112.26	1
10-Dec-14	47.58	47.66	46.7	46.9	30431788	10-Dec-14	535.9	539.36	527.49	528.04	2316204	10-Dec-14	312	313.19	304.68	305.84	3245890	10-Dec-14	114.41	11
9-Dec-14	47.11	47.92	47.05	47.59	24330506	9-Dec-14	525.88	536.65	523.41	536.11	2168813	9-Dec-14	302.99	313.64	301.14	312.5	4049506	9-Dec-14	110.19	1
8-Dec-14	48.26	48.35	47.44	47.7	26663107	8-Dec-14	529.22	533.82	527	530.73	3231818	8-Dec-14	311.57	316.56	304.82	306.64	3639180	8-Dec-14	114.1	11
5-Dec-14	48.82	48.97	48.38	48.42	27313449	5-Dec-14	536.7	538.2	527.26	528.08	3070118	5-Dec-14	316.8	316.93	310.84	312.63	3265214	5-Dec-14	115.99	11
4-Dec-14	48.39	49.06	48.2	48.84	30345132	4-Dec-14	537.64	542.69	534.89	542.58	1633688	4-Dec-14	315.53	318.59	313.47	316.93	3296642	4-Dec-14	115.77	1
3-Dec-14	48.44	48.5	47.8	48.08	23534752	3-Dec-14	537.5	541.4	535.21	536.97	1623977	3-Dec-14	325.73	326.77	314.36	316.5	5689904	3-Dec-14	115.75	11
2-Dec-14	48.84	49.05	48.2	48.46	25773478	2-Dec-14	539.45	541.85	534.66	538.59	2073974	2-Dec-14	327.5	327.93	323.25	326.31	2790257	2-Dec-14	113.5	11

索引对性能的影响

数据量	全表扫描时间
~100	~3ms
~1,000	~30ms
~10,000	~300ms
~100,000	~3s
~500,000	~15s
~1,000,000	~30s

（线性增加）

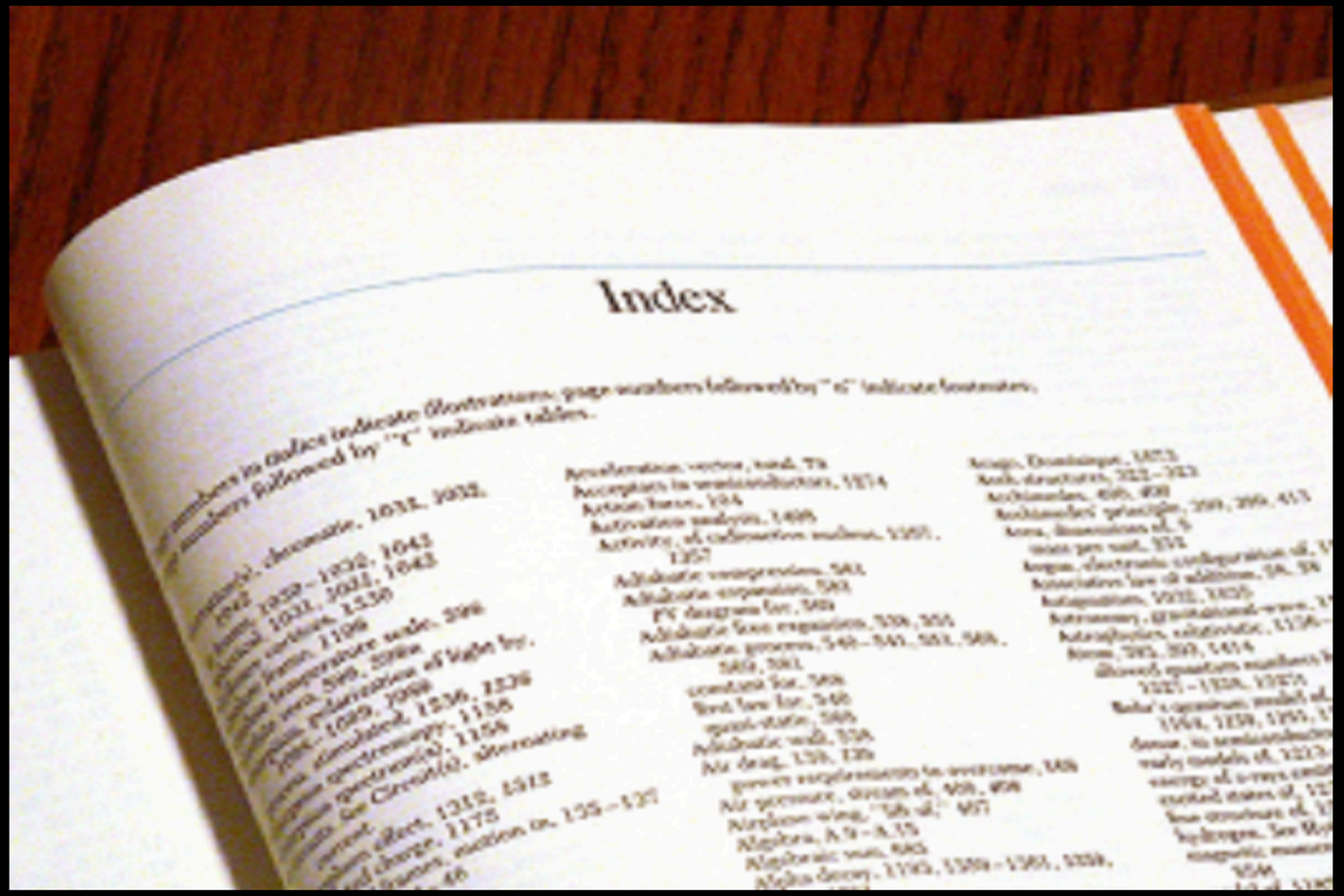
索引对性能的影响

数据量	索引查找时间
~100	~3ms
~1,000	~10ms
~10,000	~17ms
~100,000	~24ms
~500,000	~28ms
~1,000,000	~31ms

（对数增加）

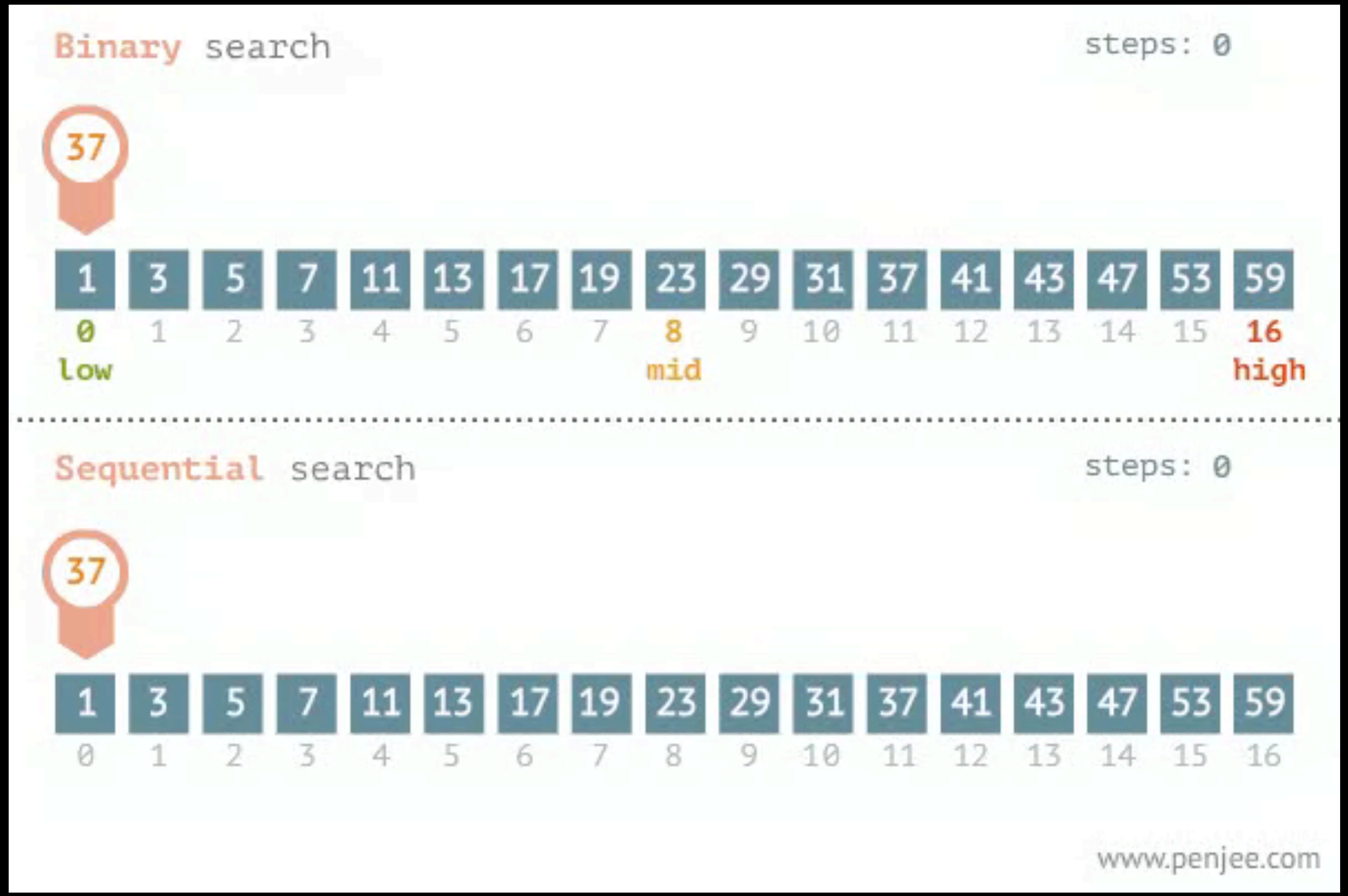
```
CREATE TABLE wp_comments (
  comment_ID bigint(20) unsigned NOT NULL auto_increment,
  //  ...
  PRIMARY KEY (comment_ID),
  KEY comment_post_ID (comment_post_ID),
  KEY comment_approved_date_gmt (comment_approved, comment_date_gmt),
  KEY comment_date_gmt (comment_date_gmt),
  KEY comment_parent (comment_parent),
  KEY comment_author_email (comment_author_email(10))
);

CREATE TABLE wp_posts (
  ID bigint(20) unsigned NOT NULL auto_increment,
  //  ...
  PRIMARY KEY  (ID),
  KEY post_name (post_name(50)),
  KEY type_status_date (post_type, post_status, post_date, ID),
  KEY post_parent (post_parent),
  KEY post_author (post_author)
);
```

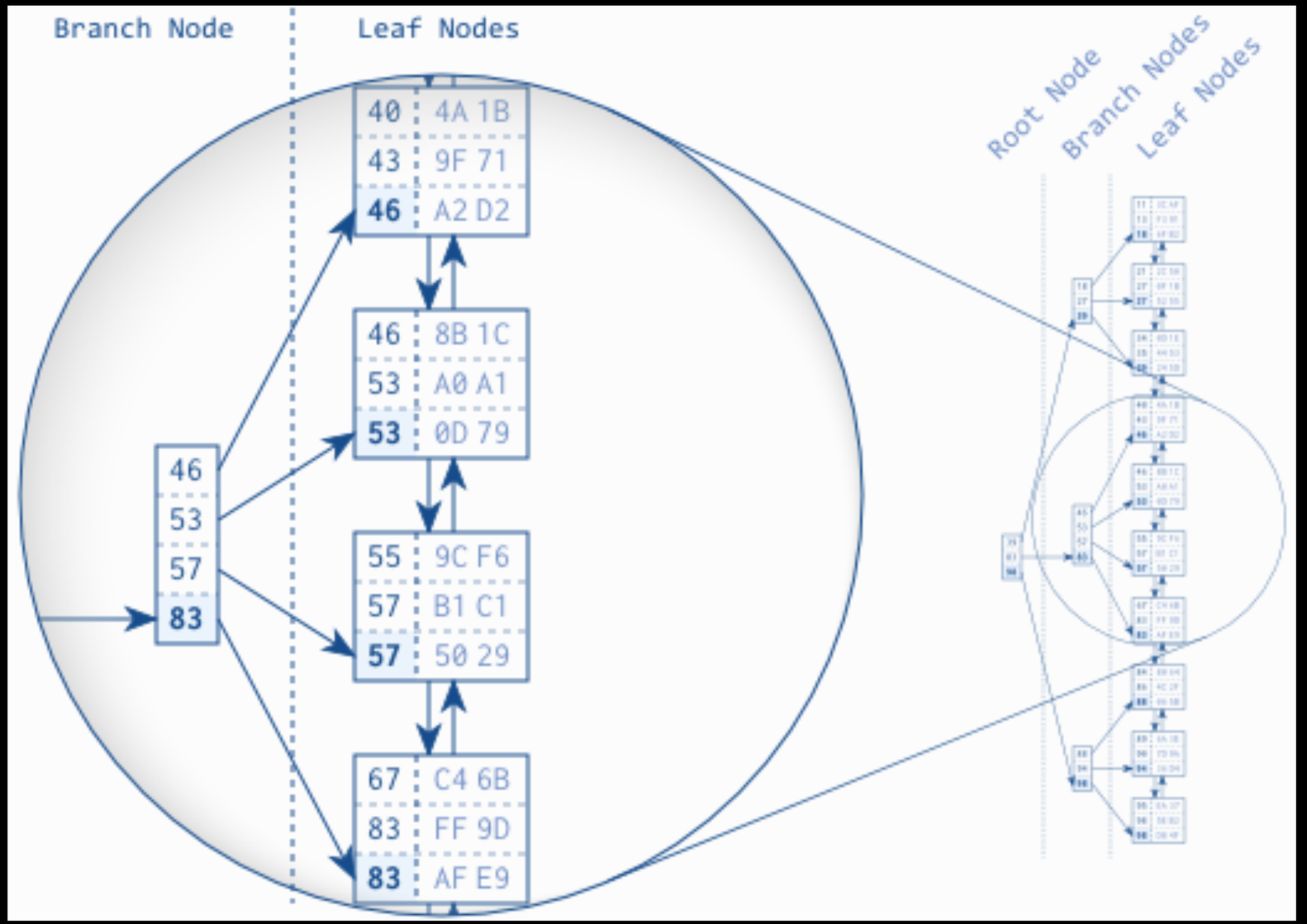


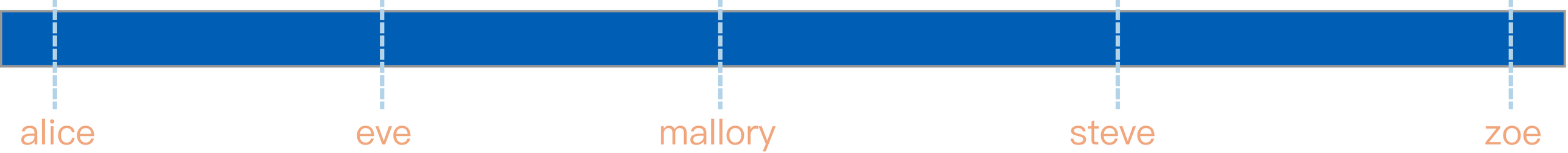
Index

二分查找

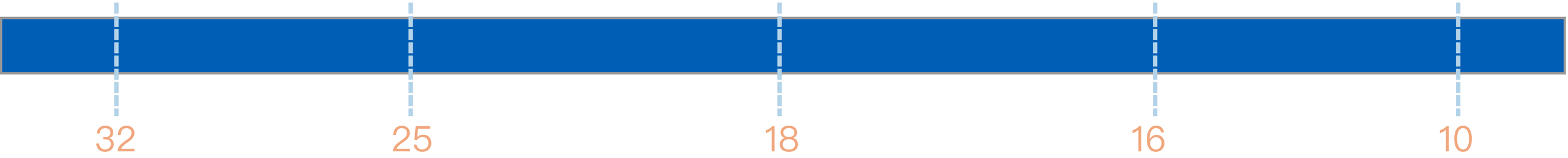

Binary search
steps: 0
37
1 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
low
mid
high
Sequential search
steps: 0
37
1 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
www.penjee.com

B-Tree


Branch Node
Leaf Nodes
46
53
57
83
40 4A 1B
43 9F 71
46 A2 D2
46 8B 1C
53 A0 A1
53 0D 79
55 9C F6
57 B1 C1
57 50 29
67 C4 6B
83 FF 9D
83 AF E9
Root Node
Branch Nodes
Leaf Nodes



{name: 1}



{age: -1}

```
select * from _User where name = 'steve'
select * from _User where name like 'ma%'

select * from _User where age > 18
select * from _User where age > 18 and age < 22
select * from _User order by age
select * from _User order by age desc
```

单列索引

- √ 全匹配查询
- √ 包含查询
- √ 前缀匹配模糊查询
- √ 大于、小于等范围查询
- √ 正、反排序
- X 后缀匹配模糊查询
- X 不等于查询

创建索引

```
CREATE INDEX idx_name ON _User (`name`);
CREATE INDEX idx_age ON _User (`age`);

db._User.createIndex({ name: 1 });
db._User.createIndex({ age: 1 });
```

索引的属性

- 主键（Primary）索引、聚集（Clustered）索引
- 唯一（Unique）索引
- 稀疏（Sparse）索引、局部（Partial）索引

区分度

```
{name: 'alice', age: 20, gender: 'female'}
{name: 'bob', age: 20, gender: 'male'}
{name: 'eve', age: 18, gender: 'female'}
{name: 'mallory', age: 29, gender: 'female'}
{name: 'steve', age: 28, gender: 'male'}
{name: 'justin', age: 28, gender: 'male'}
```

```
select * from _User where name = ? and gender = ?
```

```
SELECT COUNT(DISTINCT name) / COUNT(*) FROM _User;
```

数据库性能优化之索引

（下半部分：复合索引、常见慢查询、优化思路）