

腾讯严选课程

Tu/ing
College
图灵学院
致敬大师
致敬未来的你

深入理解Mysql索引 底层数据结构与算法

诸葛老师

十余年一线研发经验
前唯品会京东资深架构师
主导多个千万级并发互联网产品研发

课程内容

- 1、索引数据结构红黑树，Hash，B+树详解
- 2、千万级数据表如何用B+树索引快速查找
- 3、聚集索引&聚簇索引&稀疏索引到底是什么
- 4、为什么DBA总推荐使用自增主键做索引
- 5、联合索引底层数据结构又是怎样的
- 6、Mysql最左前缀优化原则是怎么回事



📍 腾讯课堂-图灵学院

🕒 7月17日 20:00



诸葛老师

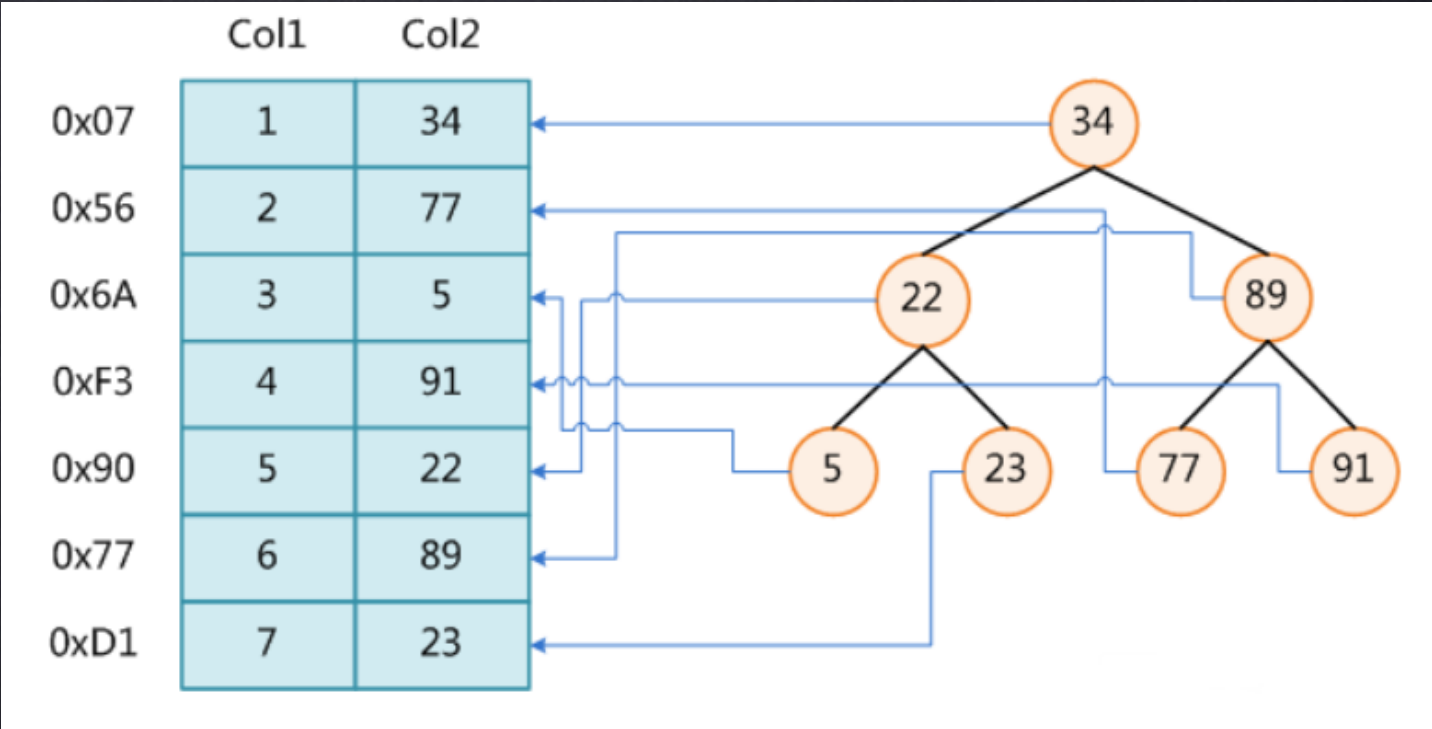
前唯品会，京东Java架构师

十余年一线互联网公司研发经验

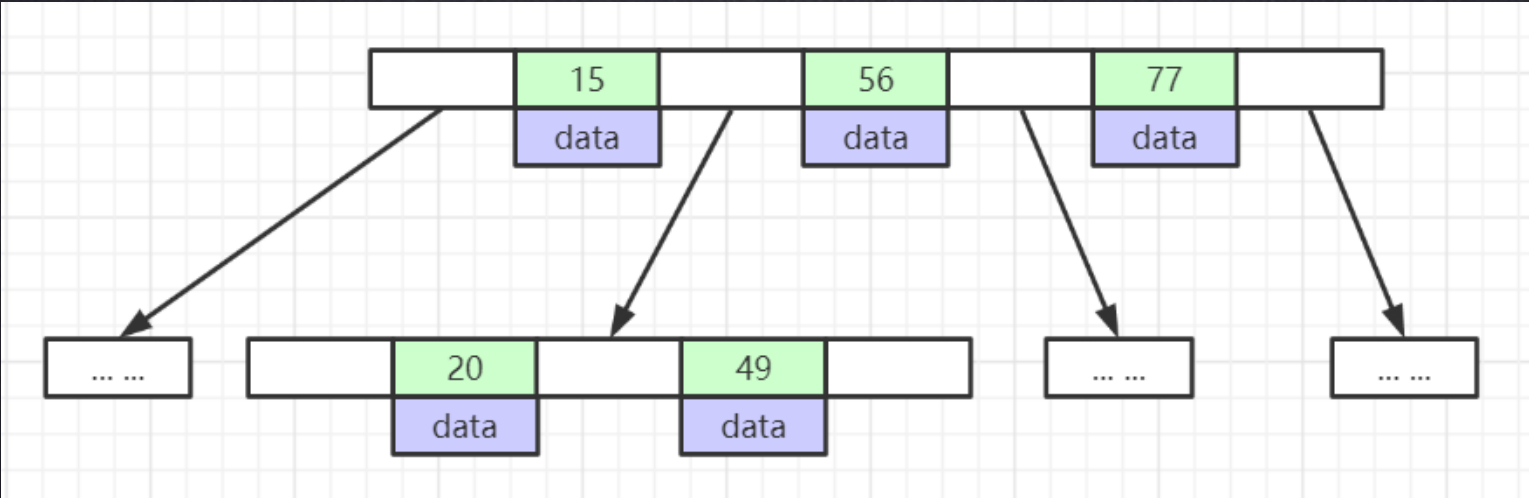
参与并主导多个千万级并发互联网项目

擅长分布式，高并发及微服务架构

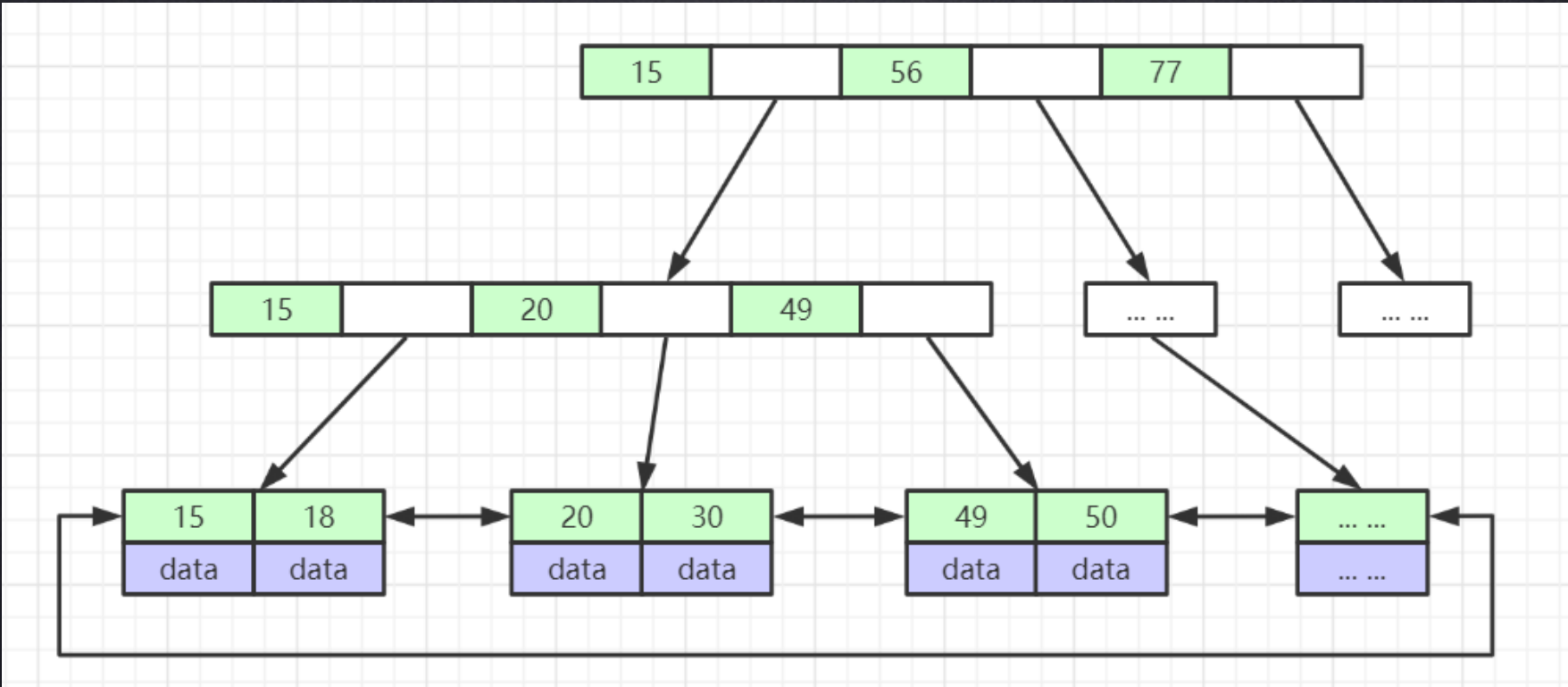
- 索引是帮助MySQL高效获取数据的**排好序的数据结构**
- 索引数据结构
 - 二叉树
 - 红黑树
 - Hash表
 - B-Tree



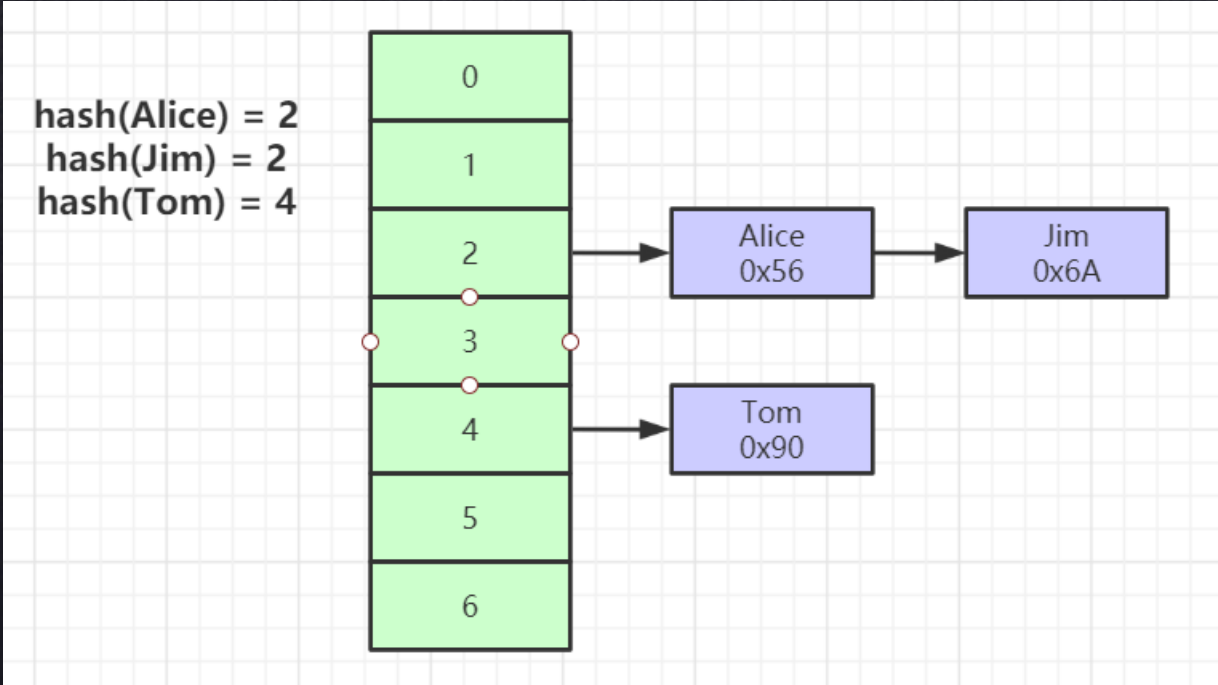
- **B-Tree**
 - 叶节点具有相同的深度，叶节点的指针为空
 - 所有索引元素不重复
 - 节点中的数据索引从左到右递增排列



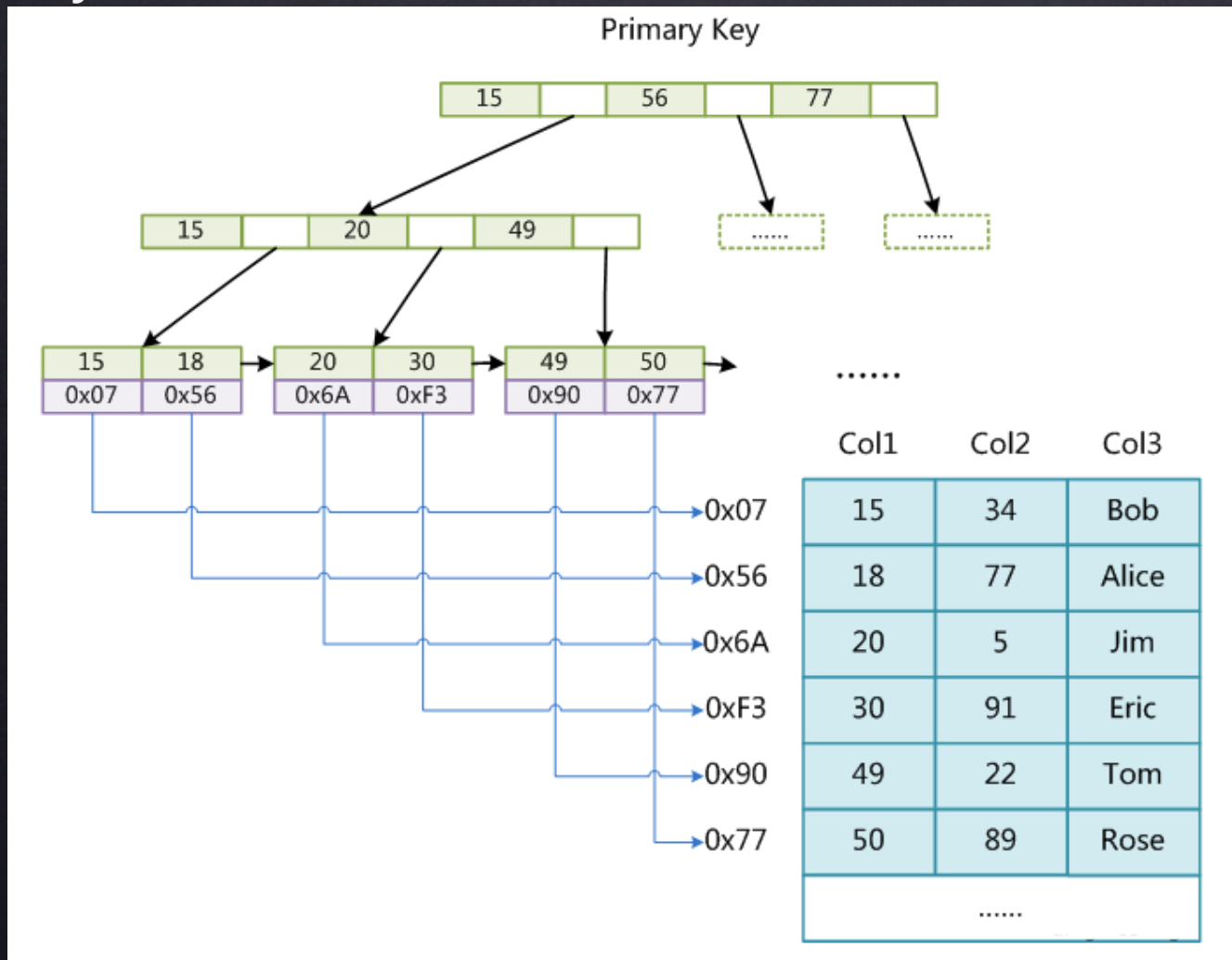
- **B+Tree**(B-Tree变种)
 - 非叶子节点不存储data，只存储索引(冗余)，可以放更多的索引
 - 叶子节点包含所有索引字段
 - 叶子节点用指针连接，提高区间访问的性能



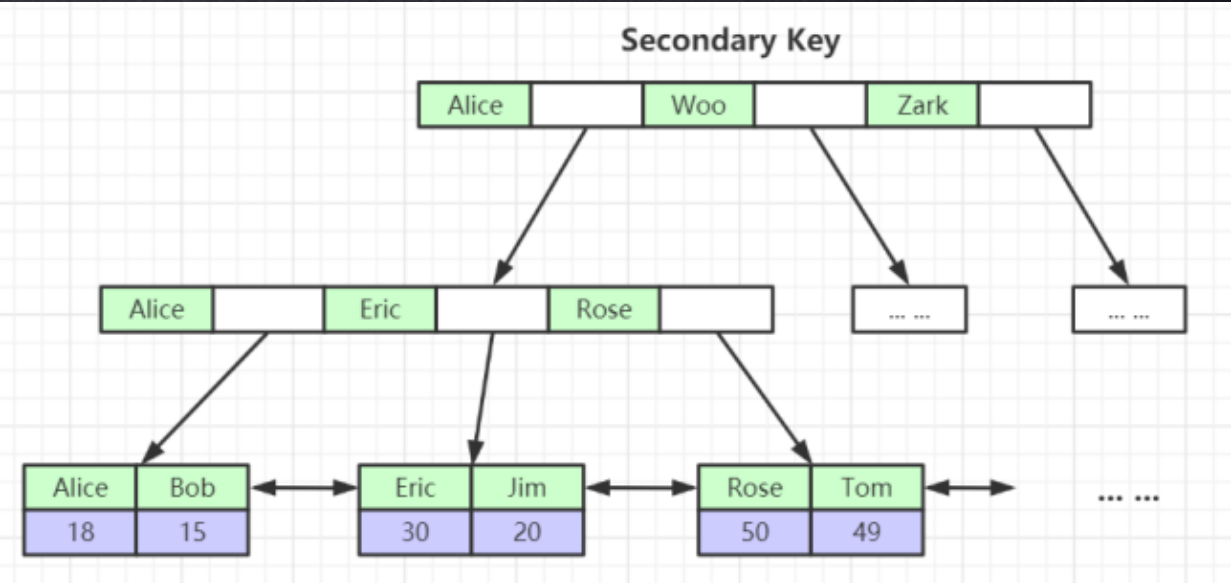
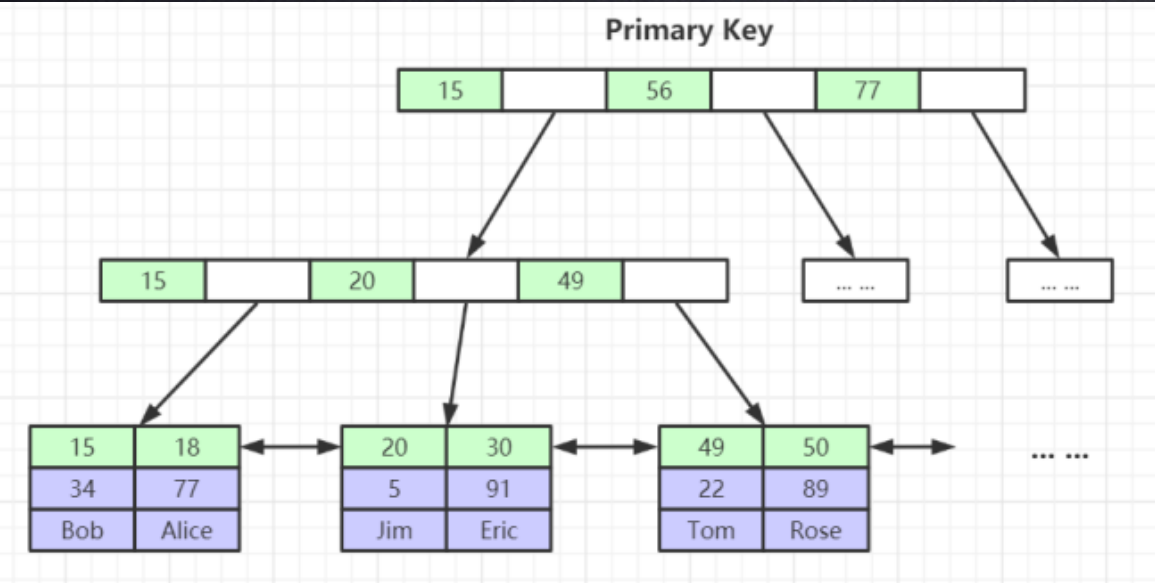
- Hash
- 对索引的key进行一次hash计算就可以定位出数据存储的位置
 - 很多时候Hash索引要比B+ 树索引更高效
 - 仅能满足 “=”, “IN”, 不支持范围查询
 - hash冲突问题



- MyISAM索引文件和数据文件是分离的(非聚集)



- InnoDB索引实现(聚集)
 - 表数据文件本身就是按B+Tree组织的一个索引结构文件
 - 聚集索引-叶节点包含了完整的数据记录
 - 为什么建议InnoDB表必须建主键，并且推荐使用整型的自增主键？
 - 为什么非主键索引结构叶子节点存储的是主键值？(一致性和节省存储空间)



- 联合索引的底层存储结构长什么样？

KEY `idx_name_age_position` (`name`,`age`,`position`) USING BTREE

