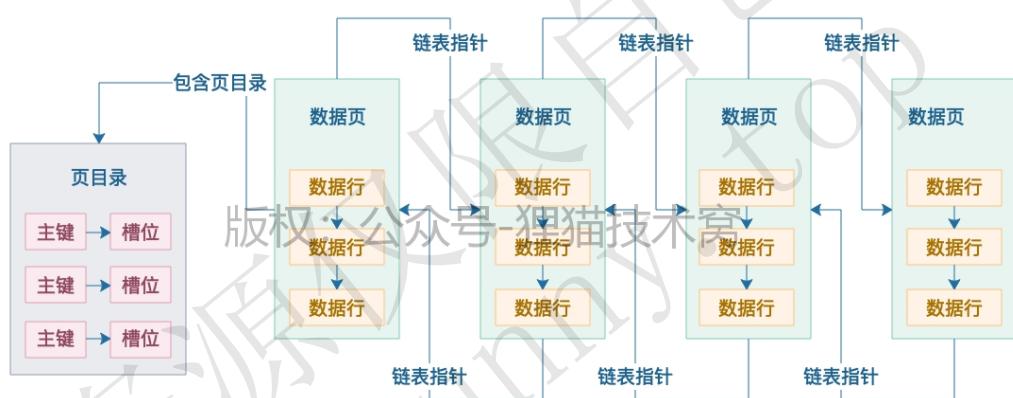


65 假设没有任何索引，数据库是如何根据查询语句搜索数据的？

假设没有任何索引，数据库是如何根据查询语句搜索数据的？

上一次我们给大家讲解了数据页在磁盘文件中的物理存储结构，大家应该目前都知道数据页之间是组成双向链表的，然后数据页内部的数据行是组成单向链表的，而且数据行是根据主键从小到大排序的。

然后每个数据页里都会有一个页目录，里面根据数据行的主键存放了一个目录，同时数据行是被分散存储到不同的槽位里去的，所以实际上每个数据页的目录里，就是这个页里每个主键跟所在槽位的映射关系，如下图所示。



所以假设你要根据主键查找一条数据，而且假设此时你数据库里那个表就没几条数据，那个表总共就一个数据页，那么就太简单了！首先就会先到数据页的页目录里根据主键进行二分查找（PS：不知道二分查找是什么的同学，建议去网上查一下，这是大学最基础算法）

然后通过二分查找在目录里迅速定位到主键对应的数据是在哪个槽位里，然后到那个槽位里去，遍历槽位里每一行数据，就能快速找到那个主键对应的数据了。每个槽位里都有一组数据行，你就是在里面遍历查找就可以了。

但是假设你要是根据非主键的其他字段查找数据呢？

那就尴尬了，此时你是没办法使用主键的那种页目录来二分查找的，只能进入到数据页里，根据单向链表依次遍历查找数据了，这就性能很差了。

好，那么现在假如我们有很多数据页呢？

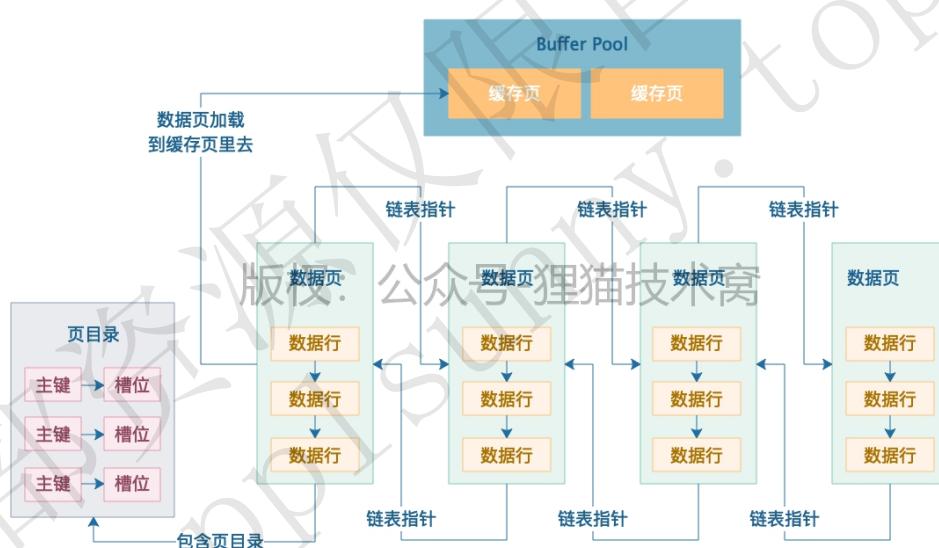
对了，一个表里往往都是有大量数据的，可能有多达成百上千个数据页，这些数据页就存放在物理磁盘文件里

所以此时是如何查询数据的呢？

之前就有不少同学在后台评论区提问过这个问题，这里我们可以先给大家解释一下，假设你要是没有建立任何索引，那么无论是根据主键查询，还是根据其他字段来条件查询，实际上都没有什么取巧的办法。

你一个表里所有数据页都是组成双向链表的吧？好，有链表就好办了，直接从第一个数据页开始遍历所有数据页，从第一个数据页开始，你得先把第一个数据页从磁盘上读取到内存buffer pool的缓存页里来。

然后你就在第一个数据页对应的缓存页里，按照上述办法查找，假设是根据主键查找的，你可以在数据页的页目录里二分查找，假设你要是根据其他字段查找的，只能是根据数据页内部的单向链表来遍历查找，如下图。



那么假设如上图所示，假设第一个数据页没找到你要的那条数据呢？

没办法，只能根据数据页的双向链表去找下一个数据页，然后读取到buffer pool的缓存页里去，然后按一样的方法在一个缓存页内部查找那条数据。

如果依然还是查找不到呢？

那只能根据双向链表继续加载下一个数据页到缓存页里来了，以此类推，循环往复。

不知道大家看到这个过程有什么感想没有？有没有觉得，你似乎是在做一个数据库里很尴尬的操作：全表扫描？

对了，其实上述操作过程，就是全表扫描，在你没有任何索引数据结构的时候，无论如何查找数据，说白了都是一个全表扫描的过程，就是根据双向链表依次把磁盘上的数据页加载到缓存页里去，然后在一个缓存页内部来查找那条数据。

最坏的情况下，你就得把所有数据页里的每条数据都得遍历一遍，才能找到你需要的那条数据，这就是全表扫描！

所以大家看完今天这篇文章，接下来我们才能正式进入索引的讲解，你才能体会到有了索引之后，是如何提升数据库的查询效率和性能的！

End