

## 图文 31 表空间以及划分多个数据页的数据区，又是什么概念？

196 人次阅读

2020-02-27 10:46:15

[返回](#)  
[前进](#)  
[重新加载](#)  
[打印](#)[详情](#) [评论](#)

## 表空间以及划分多个数据页的数据区，又是什么概念？

**如何提问：**每篇文章都有评论区，大家可以尽情留言提问，我会逐一答疑**如何加群：**购买狸猫技术窝专栏的小伙伴都可以加入狸猫技术交流群，一个非常纯粹的技术交流的地方

具体加群方式，请参见目录菜单下的文档：《MySQL专栏付费用户如何加群》（购买后可见）



狸猫技术窝

[进店逛](#)

上一次我们讲完了数据页的具体存储结构，当然里面有很多的细节我们还没讲，实际上现在也确实没必要去说那些细节，因为很多数据页的一些细节性的东西，都是要在后续讲解的内容中涉及到的，比如说数据的删除，查询的一些原理。

现在我们在大致了解了数据页的结构和使用之后，我们可以继续来了解下一个概念，就是表空间和数据区的概念

首先我们先说一下，什么是表空间？

简单来说，就是我们平时创建的那些表，其实都是有一个表空间的概念，在磁盘上都会对应着“表名.ibd”这样的磁盘数据文件

所以其实在物理层面，表空间就是对应一些磁盘上的数据文件。

有的表空间，比如系统表空间可能对应的是多个磁盘文件，有的我们自己创建的表对应的表空间可能就是对应了一个“表名.ibd”数据文件。

然后在表空间的磁盘文件里，其实会有很多很多的数据页，因为大家都知道一个数据页不过就是16kb而已，总不可能一个数据页就是一个磁盘文件吧。

所以一个表空间的磁盘文件里，其实是有很多的数据页的。

但是现在有一个问题，就是一个表空间里包含的数据页实在是太多了，不便于管理，所以在表空间里又引入了一个**数据区**的概念，英文就是**extent**

一个数据区对应着连续的64个数据页，每个数据页是16kb，所以一个数据区是1mb，然后256个数据区被划分为了一组。

对于表空间而言，他的第一组数据区的第一个数据区的前3个数据页，都是固定的，里面存放了一些描述性的数据。比如FSP\_HDR这个数据页，他里面就存放了表空间和这一组数据区的一些属性。

IBUF\_BITMAP数据页，里面存放的是这一组数据页的所有insert buffer的一些信息。

INODE数据页，这里也是存放了一些特殊的信息

## 相关频道

从零开始  
实战优化  
已更新3

大家暂时先不用了解这些东西具体是干什么的，你只要知道每一个组数据区的第一个数据区的前3个数据页，都是存放一些特殊的信息的。

然后这个表空间里的其他各组数据区，每一组数据区的第一个数据区的头两个数据页，都是存放特殊信息的，比如XDES数据页就是用来存放这一组数据区的一些相关属性的，其实就是很多描述这组数据区的东西，现在大家也不用去知道是什么。

其实今天的内容讲到这里就差不多了，讲太多大家可能就被绕晕了，大家只要知道，**我们平时创建的那些表都是有对应的表空间的，每个表空间就是对应了磁盘上的数据文件，在表空间里有很多组数据区，一组数据区是256个数据区，每个数据区包含了64个数据页，是1mb**

然后表空间的第一组数据区的第一个数据区的头三个数据页，都是存放特殊信息的；

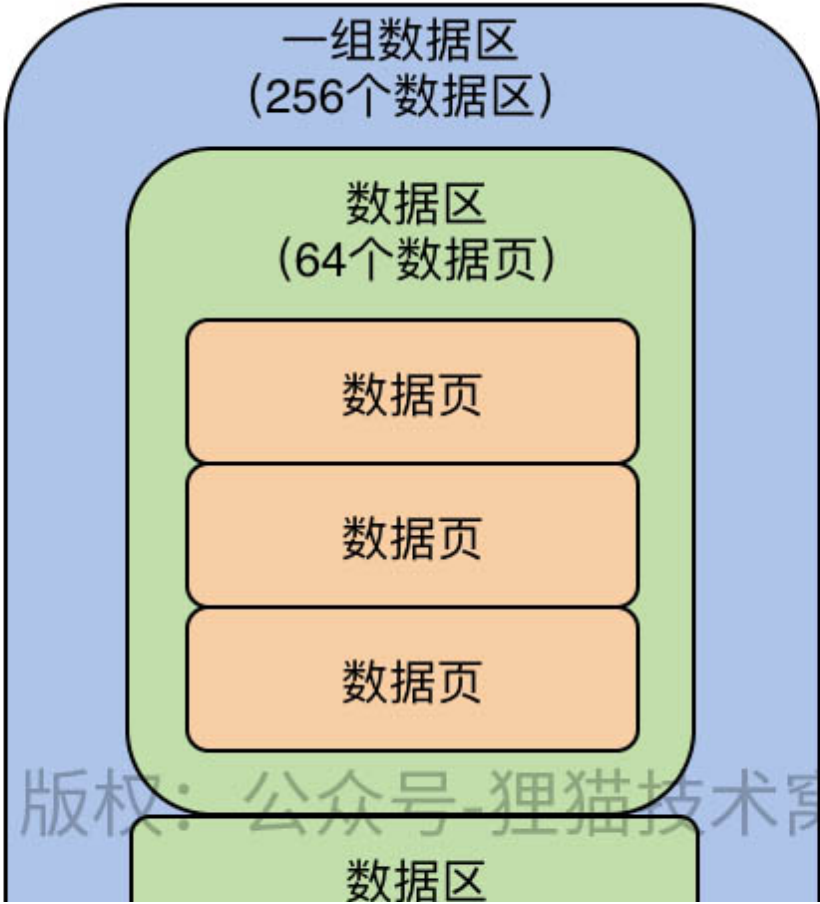
表空间的其他组数据区的第一个数据区的头两个数据页，也都是存放特殊信息的。大家今天只要了解到这个程度就可以了。

所以磁盘上的各个表空间的数据文件里是通过数据区的概念，划分了很多很多的数据页的，因此**当我们需要执行crud操作的时候，说白了，就是从磁盘上的表空间的数据文件里，去加载一些数据页出来到Buffer Pool的缓存页里去使用。**

我下面给出了一张图，图里就给出了一个表空间内部的存储结构，包括一组一组的数据区，每一组数据区是256个数据区，然后一个数据区是64个数据页。

请大家牢记下图：

返回  
前进  
重新加载  
打印



返回  
前进  
重新加载  
打印