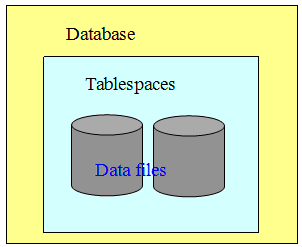
[Oracle表空间（tablespaces）](http://www.cnblogs.com/fnng/archive/2012/08/12/2634485.html)

2012-08-12 14:35 by 虫师, 23093 阅读, 4 评论, [收藏](http://www.cnblogs.com/fnng/archive/2012/08/12/2634485.html), [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=2634485)

　　我们知道oarcle数据库真正存放数据的是数据文件（data files），Oarcle表空间（tablespaces）实际上是一个逻辑的概念，他在物理上是并不存在的，那么把一组data files 捻在一起就成为一个表空间。



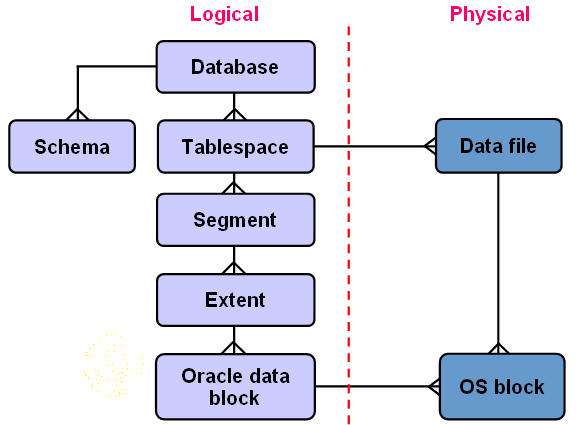
**表空间属性：**

一个数据库可以包含多个表空间，一个表空间只能属于一个数据库

一个表空间包含多个数据文件，一个数据文件只能属于一个表空间

表这空间可以划分成更细的逻辑存储单元

Oracle数据库的存储结构：



　　从逻辑的角度来看，一个数据库（database）下面可以分多个表空间（tablespace）；一个表空间下面又可以分多个段（segment）；一个数据表要占一个段（segment），一个索引也要占一个段（segment ）。 一个段（segment）由多个 区间（extent）组成，那么一个区间又由一组连续的数据块（data block）组成。这连续的数据块是在逻辑上是连续的，有可能在物理磁盘上是分散。

　　那么从物理的角度上看，一个表空间由多个数据文件组成，数据文件是实实在在存在的磁盘上的文件。这些文件是由oracle数据库操作系统的block 组成的。

**Segment（段）** ：段是指占用数据文件空间的通称，或数据库对象使用的空间的集合；段可以有表段、索引段、回滚段、临时段和高速缓存段等。

**Extent （区间）**：分配给对象（如表）的任何连续块叫区间；区间也叫扩展，因为当它用完已经分配的区间后，再有新的记录插入就必须在分配新的区间（即扩展一些块）；一旦区间分配给某个对象（表、索引及簇），则该区间就不能再分配给其它的对象.

查看表空间：

[复制代码](javascript:void(0);)

　　SQL> select \* from v$tablespace;

　　 TS# NAME INCLUD BIGFIL FLASHB ENCRYP

　　---------- ------------------------------------------------------------ ------------ ------ ------

　　 0 SYSTEM YES NO YES

　　 1 UNDOTBS1 YES NO YES

　　 2 SYSAUX YES NO YES

4 USERS YES NO YES

　　 3 TEMP NO NO YES

[复制代码](javascript:void(0);)

查看每个表空间有哪些数据文件：

[复制代码](javascript:void(0);)

　　SQL> desc dba\_data\_files;

　　 Name Null? Type

　　 ----------------------------------------- -------- ----------------------------

　　 FILE\_NAME VARCHAR2(513)

　　 FILE\_ID NUMBER

　　 TABLESPACE\_NAME VARCHAR2(30)

　　 BYTES NUMBER

　　 BLOCKS NUMBER

　　 STATUS VARCHAR2(9)

　　 RELATIVE\_FNO NUMBER

　　 AUTOEXTENSIBLE VARCHAR2(3)

　　 MAXBYTES NUMBER

　　 MAXBLOCKS NUMBER

　　 INCREMENT\_BY NUMBER

　　 USER\_BYTES NUMBER

　　 USER\_BLOCKS NUMBER

　　 ONLINE\_STATUS VARCHAR2(7)

[复制代码](javascript:void(0);)

查看详细数据文件：

[复制代码](javascript:void(0);)

SQL> select file\_name,tablespace\_name from dba\_data\_files;

　　FILE\_NAME TABLESPACE\_NAME

　　----------------------------------------------------------------------------------------------------------------

　　/ora10/product/oradata/ora10/users01.dbf USERS

　　/ora10/product/oradata/ora10/sysaux01.dbf SYSAUX

　　/ora10/product/oradata/ora10/undotbs01.dbf UNDOTBS1

/ora10/product/oradata/ora10/system01.dbf SYSTEM

[复制代码](javascript:void(0);)

创建一个表空间：

　SQL> create tablespace paul datafile '/ora10/product/oradata/ora10/paul01.dbf' size 20m;

Tablespace created.

查看我们创建的表空间：

[复制代码](javascript:void(0);)

[ora10@localhost ora10]$ pwd

　　/ora10/product/oradata/ora10

　　[ora10@localhost ora10]$ ls

　　control01.ctl control03.ctl redo01.log redo03.log system01.dbf undotbs01.dbf

　　control02.ctl paul01.dbf redo02.log sysaux01.dbf temp01.dbf users01.dbf

[复制代码](javascript:void(0);)

表空间根据对区间（extents）的管理分为两种类型

**词典管理表空间**(Dictionary-managed tablespaces)

    在表空间里，有的区间被占用了，有的没被占用，这些数据是放在数据字典里的。当你对这个表空间进行分配或释放的时候，数据文件里相关的表就会做修改。

**本地管理表空间**（locally managed tablespace）

      本地管理表空间不是在数据词典里存储表空间的，由自由区管理的表空间。用位图来自由的管理区间。一个区间对一个位，如果这个位是1表示已经被占用，0表示未被占用。

　　词典管理空间表示“中央集权治”，本地管理表空间表示“省市自治区”，一个databases表示中国，tablespaces表示一个省或直辖市。词典管理统一由中央调配。而本地管理表示有高度的自治权利，自已各种资源的分配不用上报中央。

**Undo  tablespace**

　　Undo 类型的表空间，当你对一张表或一条记录进行修改的时候，它会对修改之前的信息进行保存，这样可以保证数据的回滚。Undo 只包含undo类型的对象，不能包含任何其他对象，只适合于数据文件和区间管理。

创建undo 类型的表空间：

 SQL>create undo tablespace  undo1 datafile '/ora10/product/oradata/ora10/paul01.dbf' size 20m;

**Temporary  Tablespaces**

 　　临时表空间，相当于一个临时的垃圾场。用于排序操作，比如你要做一次大数据量的查询，但在内存无法存储这么大量的数据，然后会在磁盘上建立一个临时的表空间用记存放这些数据。Oracle就会用这个临时表空间做排序，存储中间结果。

一个全局的临时表空间，可以由多个用户共享，谁需要谁使用。但它只能存放临时的数据，不能包含任何永久性对象。 建议用本地管理方式创建这个表空间。

创建临时表空间：

 SQL>create temporary tablespace  temp datafile '/ora10/product/oradata/ora10/paul01.dbf' size 20m  extent management local uniform size 4m;

**删除表空间：**

删除表空间，使用命令drop tablespace ‘表空间名’  但是有3个选项需要注意：   
  
INCLUDING CONTENTS:指删除表空间中的segments；   
  
INCLUDING CONTENTS AND DATAFILES:指删除segments和datafiles；   
  
CASCADE CONSTRAINTS:删除所有与该空间相关的完整性约束条件。

例：

DROP TABLESPACE FESCO ‘表空间名’ CONTENTS AND DATAFILES CASCADE CONSTRAINTS;