

第 6 章 逻辑存储结构

Oracle 与 SQL Server 数据库中的逻辑存储结构是指在操作系统上看不到，只能在数据库中看到和操作的存储结构。Oracle 数据库的逻辑存储结构包括表空间(tablespace)、段(segment)、区(extent)、数据块(data block)，SQL Server 中对应的逻辑存储结构为文件组(file group)、区(extent)、数据页(data page)。区是分配存储空间单位，数据块、数据页是读取数据的单位，也是最小的逻辑存储单位。

本章主要包括：

- 表空间与文件组的功能分类
- 默认表空间与默认文件组
- 数据文件
- 空间分配单位
- 最小存储单位
- Oracle 的 segment
- 表空间及文件组管理
- 修改数据文件属性
- 移动数据文件
- 查询表空间或文件组信息
- 查询数据文件信息
- 查询表分配到的 extent 信息

6.1 表空间与文件组的功能分类

Oracle 的表空间在 SQL Server 中对应的概念是文件组，都由数据文件构成，用于存储表或索引等数据库对象。

6.1.1 分类与功能对比

Oracle 数据库的表空间分为 system 表空间、sysaux 表空间、undo 表空间、临时表空间以及用户表空间，其中 system 与 sysaux 表空间不能删除，其名称也不能修改。system 表空间主要用于存放数据字典等系统信息，undo 表空间用于存放修改数据时的多版本数据，临时表空间用于存放排序或散列操作所产生的临时数据。undo 表空间与临时表空间只能由 Oracle 使用，用户不能把表或索引存入这两个表空间。

SQL Server 的文件组分为 primary 文件组与用户文件组，分别对应 Oracle 数据库中的 system 表空间与用户表空间。与 Oracle 数据库的 system 表空间类似，primary 文件组不能删除，其名称也不能修改。SQL Server 数据库中没有对应于 Oracle 的临时表空间的文件组，SQL Server 的多版本数据以及排序或散列操作所产生的临时数据都存储于 tempdb 系统数据库中，多个数据库共用 tempdb 数据库。

6.1.2 表空间与文件组的对应关系

我们用下面的表格对 Oracle 的表空间与 SQL Server 的文件组进行一下总结。

表 6-1 表空间及文件组的对应关系

Oracle	SQL Server	功能
system 表空间	primary 文件组	存储数据库系统信息
sysaux 表空间	无	存储一些工具使用的数据
undo 表空间	tempdb 数据库	存储多版本数据
临时表空间		存储临时数据
用户表空间	用户文件组	存储用户数据

6.2 默认表空间与默认文件组

用户创建表或索引时，若未指定表空间和文件组，则存入默认表空间和文件组。

6.2.1 Oracle 数据库的默认表空间

Oracle 数据库有数据库默认表空间及用户默认表空间的概念，用户默认表空间作为用户的属性可以在创建数据库用户时指定，也可以在创建用户后修改。用户创建表或索引时，若未指定表空间，则对象存入其默认表空间，如果对用户未指定默认表空间，则此用户使用数据库默认表空间，数据库默认表空间在创建数据库时指定。

用户能够在某个表空间上创建表或索引，除了要具备 create table 或 create index 权限外，还要具备使用这个表空间的配额(quota)。

设置默认表空间可以使用 alter database 命令，如下面命令把 users 设置为默认表空间：

```
SQL> alter database default tablespace users;
```

查询数据库默认表空间可以使用以下命令：

```
SQL> select property_value from database_properties
2  where property_name='DEFAULT_PERMANENT_TABLESPACE'
3  /
```

```
PROPERTY_VALUE
-----
USERS
```

关于用户默认表空间及其他用户属性请参考第 11 章相关内容。

6.2.2 SQL Server 数据库的默认文件组

SQL Server 数据库的默认文件组对应于 Oracle 数据库的默认表空间，默认为 primary 文件组。SQL Server 没有用户默认文件组的概念。

SQL Server 数据库没有文件组使用配额的概念，具备相关权限的用户可以把表或索引创建至任意一个文件组。

SQL Server 也使用 alter database 命令设置数据库默认文件组，如下面命令把 fg 文件组设置为默认文件组(fg 文件组要预先存在)：

```
1> alter database law modify filegroup fg default
2> go
```

通过查询目录视图 sys.data_spaces 确认默认文件组：

```
1> select name from sys.data_spaces
2> where is_default=1
3> go
```

```
name
-----
fg
```

6.2.3 把表创建到指定表空间或文件组

Oracle 把表创建到指定表空间，只要在普通建表语句中附加 `tablespace` 子句，如下面命令把 `t` 表创建至 `tbs` 表空间：

```
SQL> create table t(a int, b int)
2 tablespace tbs
3 /
```

要查看表和表空间的关系，可以使用 `dba_tables` 数据字典视图，如查询上面 `t` 表所在的表空间：

```
SQL> select table_name, tablespace_name
2 from dba_tables
3 where table_name='T'
4 and owner='SYSTEM'
5 /
```

TABLE_NAME	TABLESPACE
T	TBS

类似地，SQL Server 使用下面命令把 `t` 表创建至 `fg` 文件组：

```
1> create table t(a int, b int)
2> on fg
```

查询表和文件组的关系，可以使用 `sys.indexes` 目录视图：

```
1> select object_name(object_id) table_name,
2>        object_schema_name(object_id) schema_name,
3>        filegroup_name(data_space_id) filegroup_name
4> from sys.indexes
5> where object_name(object_id)='t'
6> go
```

table_name	schema_name	filegroup_name
t	sch	PRIMARY
t	dbo	fg

6.3 数据文件

数据文件构成表空间或文件组，Oracle 和 SQL Server 的数据文件有很多共性，但在管理和操作方面也有一些区别。

6.3.1 为什么大型数据库一般使用多个文件存储数据

Oracle 数据库的表空间与 SQL Server 数据库的文件组一般由多个数据文件构成，其好处主要有以下两点：

- 如果一个表的数据存储于多个文件，而这多个文件又是创建于不同磁盘，则访问表的数据时，可以实现并行访问，从而提高数据读写效率。
- 如果数据库数据量很大，备份数据库时，可以以文件组或数据文件作为备份单位，

而不必总是执行全库备份。

6.3.2 Oracle 的数据文件

Oracle 数据库的表空间由若干个数据文件组成，默认扩展名为 `dbf`。数据文件以及重做日志文件的路径、名称以及空间大小等信息保存在控制文件中，启动数据库时，Oracle 通过读取控制文件得到其他文件的信息。数据库的系统信息一般存储在 `system` 表空间的第一个数据文件中，其地位与 SQL Server 的主数据文件相似。

6.3.3 SQL Server 的数据文件

与 Oracle 类似，SQL Server 的文件组也由若干个数据文件组成，`primary` 文件组的第一个数据文件称为主数据文件，其文件名称的默认扩展名为 `mdf`，数据库中的所有其他数据文件都称为辅助数据文件，文件名称的默认扩展名为 `ndf`。

主数据文件与辅助数据文件的主要区别在于：除了存储系统以及用户数据以外，主数据文件中还存储了数据库中的所有辅助数据文件以及重做日志文件的路径、名称、大小等信息。SQL Server 通过读取主数据文件得到其他数据文件及重做日志文件的信息，这个功能与 Oracle 的控制文件相似。

6.4 空间分配单位：extent

SQL Server 数据库的 `extent` 与 Oracle 数据库的 `extent` 功能相同，一般翻译为区，都是对表或索引分配存储空间单位，由数据文件上连续的多个数据块或数据页构成。

两者都是添加第一行记录时才对表分配区。

Oracle 的 `extent` 大小在创建表空间时确定。创建表空间时，若附加 `uniform size` 子句，此表空间的 `extent` 大小都为 `uniform size` 子句中指定的值；若附加 `autoallocate` 子句(默认选项)，则由 Oracle 自动分配，这时，此表空间上的表或索引被分配的前 16 个 `extent` 为 64kB，以后逐渐增大。

SQL Server 的 `extent` 大小是固定的 8 个数据页，64kB。创建文件组时，不能指定类似 Oracle 中的 `autoallocate` 或 `uniform size` 子句。

6.5 最小存储单位：data block 与 data page

SQL Server 的数据页与 Oracle 数据库的数据块相同，是最小的存储单位，也是读写数据的单位。

Oracle 数据库中的数据块大小默认为 8KB，还可以为 2 KB，4 KB，16 KB，32 KB。其默认数据块大小由 `db_block_size` 参数指定，创建表空间时，也可以根据这个表空间内存储数据的特点指定另外的数据块大小。`system`、`sysaux` 以及临时表空间的数据块大小都必须是 `db_block_size`，不能另外指定。

SQL Server 数据页大小固定为 8KB，不能修改，与 Oracle 创建表空间时可以指定另外的数据块大小不同。

6.6 Oracle 的 segment

与 SQL Server 不同, Oracle 还存在另外一种逻辑存储概念称为 **segment**, 一般翻译为段, 每个段对应 Oracle 作为整体处理的一个数据库对象, 由若干个 **extent** 组成。

Oracle 数据库中的段分为: 数据段, 索引段, 临时段以及 **undo** 段。

数据库中最重要对象显然是表, 每个普通表在数据字典中都对应一个同名的数据段, 但是段并不是另外一种对象, 还是表本身, 或者说表存储在其段中。把表称为段关注的是其存储空间, 如分配给这个表的 **extent**、**data block** 等信息, 称其为表关注的是其中的逻辑数据, 即其行与列等数据。表及其段可以看做是一种事物的两种叫法, 不同叫法关注的方面不同而已。分区表(**partitioned table**)所包含的每个分区(**partition**)对应一个段, 聚簇表(**clustered table**)可以包含多个段。

类似地, 索引段对应数据库中一个索引。

当内存容量不足以完成排序、散列等操作时, Oracle 会在临时表空间分配临时段存放临时数据, 操作完成后, 其占用的临时段也会随之释放。

undo 段存储 DML 操作的前映像(**before image**), 如 **update** 操作执行之前的数据先被保存至 **undo** 段, 然后再修改数据, 这些之前的旧数据即为前映像。**undo** 段提供以下功能:

- 提供读一致性, 即当某个事务未提交时, 其他事务只能读取其开始之前的旧数据。
- 用户回滚一个事务时, 用这些旧数据替换修改过的数据。
- 执行数据库实例恢复时(即由于断电或数据库崩溃而导致数据库重启时, Oracle 自动执行的恢复操作), 自动回滚未提交的事务。

6.7 表空间及文件组管理

本节内容介绍 Oracle 管理表空间与 SQL Server 管理文件组的 SQL 语法。

在 Oracle 数据库中, 使用 **create tablespace tbsname** 命令创建表空间, 同时指定此表空间包含的数据文件及其属性。

在 SQL Server 中, 使用 **alter database dbname add filegroup fgroup** 向数据库添加文件组, 其中 **dbname** 及 **fgroup** 分别为数据库名称及文件组名称, 创建文件组后再对其添加数据文件, 这与 Oracle 数据库在创建表空间命令中就指定数据文件不同。

创建文件组和表空间时, 可以指定的数据文件属性, SQL Server 与 Oracle 基本相同, 都包括文件物理名称、文件大小、是否自动增长、最大大小限制等, 除此之外, SQL Server 还要指定逻辑文件名, 而 Oracle 不需要, 在 SQL Server 中修改数据文件的属性时, 只要指定逻辑文件名即可, 而不用使用物理文件名。

6.7.1 创建表空间或文件组

先说明 Oracle 创建表空间的语法形式。下面命令都以 **system** 用户执行:

```
SQL> create tablespace tbs1
2 datafile 'e:\oradata\tbs1_01.dbf' size 5m autoextend on next 1m maxsize 100m
3 uniform size 128k
4 /
```

以上命令中的第 1 行指定表空间名称, 第 2 行指定表空间所包含的数据文件的路径及文件名称, 大小, 以及自动增长和最大大小等属性, 第 3 行指定此表空间 **extent** 的分配方式为统一大小 128KB。

第 2 行的各参数含义如下：

- size: 指定文件初始大小。
- autoextend: 指定文件是否自动增长，其值为 on 或 off。
- next: 文件为自动增长时，用于设置增长的大小。
- maxsize: 文件为自动增长时，用于指定文件的最大大小限制。

下面命令创建包含两个数据文件的表空间 tbs2:

```
SQL> create tablespace tbs2
2 datafile
3 'e:\oradata\tbs2_01.dbf' size 5m autoextend on next 1m maxsize 100m,
4 'e:\oradata\tbs2_02.dbf' size 5m
5 autoallocate
6 /
```

上面命令中的第 4 行未指定数据文件 tbs2_02.dbf 的自动增长属性，则其大小不能自动增长。第 5 行指定了表空间的 autoallocate 属性，则此表空间的 extent 的大小由 Oracle 指定。

下面命令在创建表空间时，指定数据块大小为 2KB。要注意先设置 db_2k_cache_size 缓冲区大小：

```
SQL> alter system set db_2k_cache_size=10m;
```

然后可以在创建表空间时，指定数据块大小：

```
SQL> create tablespace tbs3
2 datafile 'e:\oradata\tbs3_01.dbf' size 5m
3 blocksize 2048
4 /
```

表空间创建后，可以根据需要对其添加新的数据文件，下面命令对 tbs2 表空间添加一个数据文件 tbs02.dbf，大小为 5MB：

```
SQL> alter tablespace tbs2
2 add datafile 'e:\oradata\tbs02.dbf' size 5m
3 autoextend on next 1m
4 /
```

SQL Server 不能在创建文件组时附加数据文件，只能在创建文件组后，添加数据文件。在数据库 law 中创建 fg1 文件组：

```
1> use law
2> go
已将数据库上下文更改为 'law'。
1> alter database law add filegroup fg1
2> go
```

再向文件组 fg1 添加两个数据文件：

```
1> alter database law
2> add file
3> (
4>     name = fg1_01,
5>     filename = 'e:\sqldata\fg1_01.ndf',
6>     size = 5MB,
7>     maxsize = 100MB,
8>     filegrowth = 5MB
9> ),
10> (
11>     name = fg1_02,
```

```

12>     filename = 'e:\sqldata\fg1_02.ndf',
13>     size = 5MB,
14>     maxsize = 100MB,
15>     filegrowth = 5MB
16> )
17> to filegroup fg1
18> go

```

各个参数的含义如下：

- **name:** 用于指定逻辑文件名称。改变文件属性时，只要指定这个逻辑名称即可。
- **filename:** 用于物理文件名称。指定文件的路径和名称。
- **size:** 用于指定数据文件大小。
- **maxsize:** 用于指定数据文件最大大小。
- **filegrowth:** 用于指定每次文件增长大小，如果未指定确定的值，则默认为 1MB，如果指定为 0，则数据文件不能自动增长。可以使用 MB、KB、GB、TB 或百分比 (%) 为单位指定。如果未在数量后面指定 MB、KB 或 %，则默认值为 MB。如果指定 %，则增量大小为发生增长时文件大小的指定百分比。指定的大小舍入为最接近的 64 KB 的倍数。

6.7.2 删除表空间或文件组

Oracle 使用 drop tablespace 删除表空间，下面命令删除 tbs2 表空间：

```
SQL> drop tablespace tbs2 including contents and datafiles;
```

当表空间中已经创建了表或索引时，要附加 including contents 子句，如果要连同数据文件一起删除，则还要附加 and datafiles 子句，否则在操作系统上的数据文件不会被删除。

SQL Server 删除文件组，使用 alter database 命令的 remove filegroup 子句，下面命令删除 fg1 文件组：

```

1> alter database law remove filegroup fg1
2> go

```

在 SQL Server 数据库中，不允许删除包含表或索引的文件组，这与 Oracle 不同。

6.7.3 修改表空间或文件组属性

为了便于对比，每个功能的两个命令放在一起，第一个为 Oracle 修改表空间属性，第二个为 SQL Server 修改文件组属性。

重命名表空间或文件组：

```

SQL> alter tablespace tbs1 rename to tbs2;
1> alter database law modify filegroup fg1 name=fg;

```

设置默认表空间或文件组：

```

SQL> alter database default tablespace tbs2;
1> alter database law modify filegroup fg default;

```

设置表空间文件组为只读：

```

SQL> alter tablespace tbs2 read only;
1> alter database law modify filegroup fg readonly;

```

设置表空间或文件组为可读写：

```
SQL> alter tablespace tbs2 read write;
1> alter database law modify filegroup fg readwrite;
```

Oracle 设置表空间脱机、联机:

```
SQL> alter tablespace tbs2 offline;
SQL> alter tablespace tbs2 online;
```

SQL Server 只能设置数据库为脱机, 而不能设置文件组为脱机。

6.8 修改数据文件属性

数据文件管理主要是修改数据文件的各种属性, 如文件大小、自动增长属性、最大大小以及移动数据文件。

Oracle 和 SQL Server 都使用 `alter database` 命令修改数据文件属性, SQL Server 还需要指定数据文件所在的数据库名称, 另外在其他细节方面也有很多不同。

为完成本节实验, 我们分别在 Oracle 和 SQL Server 中新建一个表空间和文件组。

Oracle 中新建表空间 `newtbs`, 包含一个数据文件 `e:\newtbs01.dbf`, 大小为 10MB:

```
SQL> create tablespace newtbs
2 datafile 'e:\newtbs01.dbf' size 10m
3 /
```

在 SQL Server 的 `law` 数据库中, 新建文件组 `newFG`, 包含一个数据文件: `e:\newFG_01.ndf`, 逻辑名称为 `newFG_01`, 大小为 5MB:

```
1> alter database law add filegroup newFG
2> go
1> alter database law add file
2> (
3>     name=newFG_01,
4>     filename='e:\newFG_01.ndf',
5>     size=5
6> )
7> to filegroup newFG
8> go
```

下面开始修改数据文件属性的各个实验, 为了便于对比, 功能相同的两个命令放在一起, 第一个为 Oracle 命令, 第二个为 SQL Server 命令。

改变数据文件大小:

```
SQL> alter database datafile 'e:\newtbs01.dbf' resize 20m;
1> alter database law modify file (name=newFG_01, size=10MB);
```

`name` 用于指定要修改的文件逻辑名称, `size` 用于指定数据文件的新大小。

改变数据文件自动增长属性:

```
SQL> alter database datafile 'e:\newtbs01.dbf' autoextend on next 1m maxsize 100m;
1> alter database law modify file (name=newFG01, filegrowth=2)
```

SQLServer 中修改数据文件逻辑名称:

```
1> alter database law modify file (name=newFG_01, newname=newFG01);
```

`name` 用于指定修改后的逻辑文件名称。

6.9 移动数据文件

因为更换磁盘，或出于磁盘分区空间限制，会有移动数据文件的需要。Oracle 和 SQL Server 都提供了移动数据文件的方法。

6.9.1 Oracle 移动数据文件

在 Oracle 12c 之前版本移动数据文件，要先把表空间脱机，然后在操作系统中移动文件。Oracle 12c 支持在线移动数据文件，只需要执行 `alter database move datafile` 命令即可。

下面命令把'e:\sqldata\ts51.dbf'移至 'd:\oradata'，并把文件名称改为 ts501.dbf：

```
SQL> alter database move datafile
2 'e:\sqldata\ts51.dbf' to 'd:\oradata\ts501.dbf'
3 /
```

若要在移动文件的同时，保留原文件，则可以附加 `keep` 选项：

```
SQL> alter database move datafile
2 'd:\oradata\ts501.dbf' to 'd:\oradata\ts501_new.dbf'
3 keep
4 /
```

6.9.2 SQL Server 移动数据文件

下面以把 law 数据库中的数据文件 e:\newFG_01.ndf 移动到 c:\newFG_01.ndf 为例，说明移动数据文件的过程。

首先把数据库脱机：

```
1> alter database law set offline
2> go
```

在操作系统中把 d:\newFG01.ndf 移动到 c:\newFG01.ndf：

```
1> !! move e:\newFG_01.ndf c:\newFG_01.ndf
```

修改数据库对此文件物理路径的记载：

```
1> alter database law modify file (name=newFG01,filename='c:\newFG_01.ndf');
```

最后再把数据库重新联机：

```
1> alter database law set online;
```

6.10 查询表空间或文件组信息

查询 Oracle 表空间信息，可以使用数据字典视图 `dba_tablespaces`，如查询数据库中所有的表空间名称，块大小及空间分配方式：

```
SQL> select tablespace_name, block_size, allocation_type from dba_tablespaces;
```

TABLESPACE_NAME	BLOCK_SIZE	ALLOCATION
SYSTEM	8192	SYSTEM
SYSAUX	8192	SYSTEM
UNDOTBS1	8192	SYSTEM

TEMP	8192 UNIFORM
USERS	8192 SYSTEM

下面命令查询数据库的默认表空间：

```
SQL> select property_value from database_properties
2  where property_name='DEFAULT_PERMANENT_TABLESPACE'
3  /
```

PROPERTY_VALUE
TBS2

对应 Oracle 数据库的 dba_tablespaces 数据字典视图, 在 SQL Server 中为 sys.filegroups, 下面命令查询 sales 数据库中所有的文件组名称及是否为默认文件组：

```
1> select name,is_default from sys.filegroups
2> go
```

name	is_default
PRIMARY	1
SalesGroup1	0
SalesGroup2	0

6.11 查询数据文件信息

Oracle 中查询数据文件信息，可以使用 dba_data_files 数据字典视图， 下面命令查询表空间 tbs3 中包含的数据文件名称、大小、是否自动增长、每次增长多少等信息：

```
SQL> select file_name, bytes/1024/1024 file_size, autoextensible, increment_by
2  from dba_data_files
3  where tablespace_name='NEWTBS'
4  /
```

FILE_NAME	FILE_SIZE	AUT	INCREMENT_BY
E:\NEWTBS01.DBF	10	NO	0

与 Oracle 数据库中的 dba_data_files 数据字典视图对应的目录视图，在 SQL Server 中为 sys.database_files 与 sys.master_files，sys.database_files 用于查询当前数据库中的数据文件及重做日志文件信息，而 sys.master_files 用于查询 master 数据库中的数据文件及重做日志文件的信息。

在这个目录视图中，关于文件组的信息，只有文件组 ID 编号，并不包含文件组名称的信息，如果要查询某个文件组内的数据文件信息，需要 sys.database_files 目录视图与 sys.filegroups 目录视图以其公共列 data_space_id 为连接条件作连接查询，另外要注意，在 sys.database_files 目录视图中除了包含数据文件信息外，还包含重做日志文件信息，这是与 Oracle 数据库的 dba_data_files 数据字典视图不同的，dba_data_files 并不包含重做日志文件的信息，重做文件信息包含在 v\$log 与 v\$logfile 这两个动态性能视图中。

下面命令查询 law 数据库中的 fg1 文件组包含的数据文件的物理名称、大小、自动增长等信息：

```
1> use law
2> go
已将数据库上下文更改为 'law'。
```

```

1> select g.name, f.physical_name, f.size*8192/1024/1024 size,
2>       f.growth*8192/1024/1024 growth
3> from sys.filegroups g, sys.database_files f
4> where g.data_space_id=f.data_space_id
5> and g.name='fg1'
6> go

```

name	physical_name	size	growth
fg1	e:\sqldata\fg1_01.ndf	5	5
fg1	e:\sqldata\fg1_02.ndf	5	5

6.11 查询表分配到的 extent 信息

在 Oracle 数据库中，要查询某个表被分配的 extent 信息，可以使用 `dba_extents` 数据字典视图。在 SQL Server 中，可以使用 `dbcc extentinfo(dbname, tname)` 命令，其中的参数 `dbname` 为要查询的表所在的数据库名称，`tname` 为要查询的表名称。

以 system 账户连接数据库，拷贝数据字典 `all_objects` 的结构及数据创建表 `t`：

```

SQL> conn system/oracle
已连接。
SQL> create table t_extents as select * from all_objects;

```

查询表 `t` 被分配的 extent 的信息：

```

SQL> select segment_name tname, tablespace_name tbsname,
2       extent_id, file_id, block_id, blocks
3 from dba_extents
4 where segment_name='T_EXTENTS' and owner='SYSTEM'
5 /

```

TNAME	TBSNAME	EXTENT_ID	FILE_ID	BLOCK_ID	BLOCKS
T_EXTENTS	SYSTEM	0	1	99224	8
T_EXTENTS	SYSTEM	1	1	99232	8
T_EXTENTS	SYSTEM	2	1	99240	8
T_EXTENTS	SYSTEM	3	1	99248	8
...					

从查询结果可以看出，表 `t` 被分配的前 4 个 extent，都在 `system` 表空间的 1 号数据文件上分配，大小为 8 个数据块，每个区的开始数据块的块号分别为 99224，99232，99240 等。

通过查询 `dba_data_files` 数据字典视图，可以得到 1 号数据文件的物理文件名称及所在路径：

```

SQL> col file_name for a100
SQL> select file_name from dba_data_files
2 where file_id=1
3 /

```

FILE_NAME
E: \ORADATA\LAU\SYSTEM01.DBF

下面在 SQL Server 中查询表的 extent 信息。

与上节内容类似，创建测试表：

```

1> use law

```

```
2> go
已将数据库上下文更改为 'law'。
1> select * into t_extents from sys.objects
2> go
```

然后使用 dbcc extentinfo 命令查询表 t_extents 被分配的 extent 信息，下面的查询结果因为版面原因，未显示所有的列：

```
1> dbcc extentinfo(law,t_extents)
2> go
```

file_id	page_id	pg_alloc	ext_size	object_id
3	96	3	8	917578307

在 dbcc extentinfo 的查询结果中，不包含 extent_id 列，不能确定 extent 的先后顺序，另外也不能选取特殊的列，这些都不如 Oracle 的 dba_extents 方便。另外，dbcc extentinfo 是 Microsoft 未公开的一个命令，在联机丛书上查不到其帮助信息。