

第 5 课

表空间

Oracle 数据库被划分为称为表空间的逻辑区域，形成 Oracle 数据库的逻辑结构，即数据库是由多个表空间构成的。Oracle 数据库的存储管理实际上是对数据库逻辑结构的管理，管理对象主要包括表空间、数据文件、段、区和数据库。在物理结构上，数据信息存储在数据文件中，而在逻辑结构上，数据库中的数据存储在表空间中。

表空间与数据文件存在紧密的对应关系，一个表空间至少包含一个数据文件，而一个数据文件只能属于一个表空间。

在 Oracle 数据库中除了基本表空间以外，还有临时表空间、大文件表空间以及还原表空间等。本课将对表空间进行详细讲解，学习各种类型表空间的创建、修改、切换和管理等操作。

本课学习目标：

- ☐ 理解表空间对逻辑结构和物理结构
- ☐ 熟练掌握创建表空间
- ☐ 掌握如何设置表空间的状态
- ☐ 了解如何重命名表空间
- ☐ 掌握表空间中数据文件的管理
- ☐ 了解临时表空间
- ☐ 了解大文件表空间
- ☐ 理解还原表空间的作用
- ☐ 掌握创建与管理还原表空间

5.1

认识 Oracle 的逻辑结构和物理结构

在第 2 课中简单介绍了 Oracle 中的逻辑结构和物理结构，本节通过这两种结构展示表空间在 Oracle 中的位置及作用。

我们知道 Oracle 数据库系统具有跨平台特性，因此在一个数据库平台上开发的数据库可以不用修改就移植到另一个操作系统平台上。因为 Oracle 不会直接操作底层操作系统的数据文件，而是提供了一个中间层，这个中间层就是 Oracle 的逻辑结构，它与操作系统无关，而中间层到数据文件的映射通过 RDMS 来完成。如图 5-1 所示了 Oracle 的中间层的这种跨平台特性。

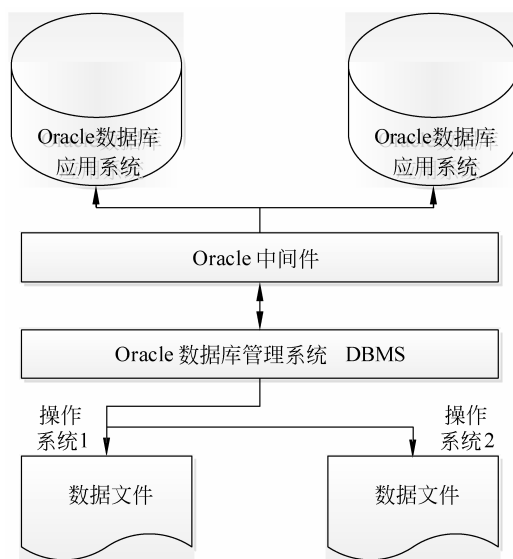


图 5-1 Oracle 跨平台特性

从图 5-1 中可以看到，Oracle 数据库应用系统通过操作中间层实现逻辑操作，而逻辑操作到数据文件操作之间是通过 DBMS 来映射完成的。这样数据文件对 Oracle 数据库应用系统是透明的。

Oracle 为了管理数据文件而引入了逻辑结构，如图 5-2 所示了数据文件管理的逻辑结构和物理结构，其中左边是逻辑结构，右边是物理结构。

1. 逻辑结构

逻辑结构从上到下是包含关系，也是一对多的关系，即一个数据库有一个或者多个表空间；一个表空间有一个或者多个段；一个段有一个或者多个区段组成；一个区段由多个数据库块组成；一个数据库块由多个操作系统数据库块组成。

2. 物理结构

在物理结构中一个表空间由一个或者多个数据文件组成，一个数据文件物理上是由操作系统块组成。

下面首先介绍图 5-2 中逻辑结构的各个组成部分。

(1) 表空间 (Table Space)

在逻辑上一个数据库由表空间组成，一个表空间只能属于一个数据库，反之不成立；一个表空间包含一个或者多个操作系统文件，这些系统文件称为数据文件。

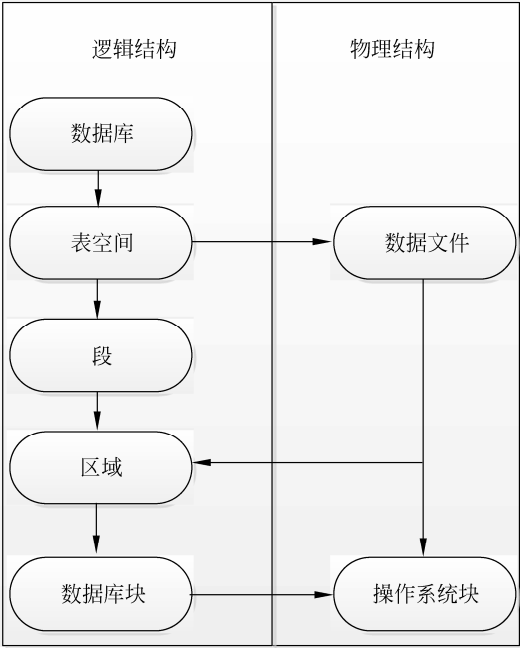


图 5-2 数据文件的逻辑结构和物理结构

(2) 段 (Segment)

段是表空间内一个逻辑存储空间，一个表空间包含一个或者多个段，一个段不能跨越表空间，即一个段只能在一个表空间中，但是段可以跨越数据文件，即一个段可以分布在同一个表空间的不同数据文件上。

(3) 区段 (Extent)

区段是段中分配的逻辑存储空间，一个区段由连续的 Oracle 数据库块组成，一个区段只能存在于一个数据文件中。在创建一个段时需要至少包含一个区段，段增长时将分配更多的区段给该段，同时 DBA 可以手动地向段中添中区段。

(4) 数据库块 (Oracle Block)

数据库块是 Oracle 数据库服务器管理存储空间中的数据文件的最小单位，也是 Oracle 数据库系统输入、输出的最小单位。一个数据库块由一个或者多个操作系统块组成。Oracle 提供了标准的数据库块尺寸，该尺寸通过初始化参数 DB_BLOCK_SIZE 设置，在初始化创建数据库时指定，一般大小为 4KB 或者 8KB。一个数据库块应该是操作系统数据块的整数倍，这样可以避免不必要的 I/O。

【练习 1】

假设我们要查看 Oracle 数据库块的大小可用如下语句。

```
SQL> select tablespace_name,block_size,contents
2  from dba_tablespaces;
```

TABLESPACE_NAME	BLOCK_SIZE	CONTENTS
SYSTEM	8192	PERMANENT
SYSAUX	8192	PERMANENT
UNDOTBS1	8192	UNDO
TEMP	8192	TEMPORARY

USERS	8192	PERMANENT
EXAMPLE	8192	PERMANENT

6 rows selected

从运行结果中可以看出，当前 Oracle 数据库中表空间的数据库大小为 8KB (8192 字节)。

物理结构的各个组成部分含义如下。

(1) 数据文件 (Data File)

数据文件是 Oracle 格式的操作系统文件，例如扩展名为 .dbf 的文件。数据文件的大小决定了表空间的大小，当表空间不足时就需要增加新的数据文件或者重新设置当前数据文件的大小，以满足表空间的增长需求。

(2) 操作系统块 (OS Block)

操作系统块依赖于不同的操作系统平台，它是操作系统操作数据文件的最小单位。一个或者多个操作系统块组成了一个数据库块。

5.2

表空间的简单操作

在创建 Oracle 数据库时系统会自动创建一系列表的表空间，使用这些表空间可以对数据进行操作，根据业务需求用户也可以创建自己的表空间。

本节首先介绍表空间的两种类型，以及表空间的四种状态属性，最后讲解创建表空间的方法。

5.2.1 表空间的分类

Oracle 中表空间的数量和大小没有严格限制，例如一个大小为 20GB 的表空间和大小为 10MB 的表空间可以并存，只是用户根据业务需求赋予的表空间功能不同。在这些表空间中有些是所有 Oracle 数据库必备的表空间，它们是 SYSTEM 表空间、临时表空间、还原表空间和默认表空间。

那些必备的表空间称为系统表空间，除此之外还有非系统表空间。

1. 系统表空间

系统表空间是 Oracle 数据库系统创建时需要的表空间，这些表空间在数据库创建时自动创建，是每个数据库必须的表空间，也是满足数据库系统运行的最低要求。例如，系统表空间 SYSTEM 中存储数据字典或者存储还原段。

在用户没有创建非系统表空间时，系统表空间可以存放用户数据或者索引，但是这样做会增加系统表空间的 I/O，影响系统效果。

2. 非系统表空间

非系统表空间是指用户根据业务需求而创建的表空间，可以按照数据多少、使用频率、需求数量等方面进行灵活地设置。这样一个表空间的功能就相对独立，在特定的数据库应用环境下可以很好地提高系统的效率。

通过创建用户自定义的表空间，如还原空间、临时表空间、数据表空间或者索引表空间，使数据库的管理更加灵活、方便。

5.2.2 表空间的状态属性

Oracle 为每个表空间都分配一个状态属性，通过设置表空间的状态属性，可以对表空间的使用进行管理。表空间状态属性有 4 种，分别是在线 (ONLINE)、离线 (OFFLINE)、只读 (READ ONLY)

和读写 (READ WRITE), 其中只读与读写状态属于在线状态的特殊情况。

1. 在线 (ONLINE)

当表空间的状态为 ONLINE 时,才允许访问该表空间中的数据。如果要将在表空间修改为 ONLINE 状态,可以使用如下语法的 ALTER TABLESPACE 语句。

```
ALTER TABLESPACE tablespace_name ONLINE;
```

2. 离线 (OFFLINE)

当表空间的状态为 OFFLINE 时,不允许访问该表空间中的数据。例如向表空间中创建表或者读取表空间中的表的数据等操作都将无法进行。这时可以对表空间进行脱机备份;也可以对应用程序进行升级和维护等。

如果要将在表空间修改为 OFFLINE 状态,可以使用如下语法的 ALTER TABLESPACE 语句。

```
ALTER TABLESPACE tablespace_name OFFLINE parameter;
```

其中, parameter 表示将表空间切换到 OFFLINE 状态时可以使用的参数,参数有如下几个选项。

(1) NORMAL

指定表空间以正常方式切换到 OFFLINE 状态。如果以这种方式切换, Oracle 会执行一次检查点,将 SGA 区中与该表空间相关的脏缓存块全部写入数据文件中,最后关闭与该表空间相关的所有数据文件。默认情况下使用此方式。

(2) TEMPORARY

指定表空间以临时方式切换到 OFFLINE 状态。如果以这种方式切换, Oracle 在执行检查点时不会检查数据文件是否可用,这会使得将该表空间的状态切换为 ONLINE 状态时,可能需要对数据库进行恢复。

(3) IMMEDIATE

指定表空间以立即方式切换到 OFFLINE 状态。如果以这种方式切换, Oracle 不会执行检查点,而是直接将表空间设置为 OFFLINE 状态,这会将该表空间的状态切换为 ONLINE 状态时,必须对数据库进行恢复。

(4) FOR RECOVER

指定表空间以恢复方式切换到 OFFLINE 状态。如果以这种方式切换,数据库管理员可以使用备份的数据文件覆盖原有的数据文件,然后再根据归档重做日志将表空间恢复到某个时间点的状态。所以此方式经常用于对表空间进行基于时间的恢复。

3. 只读 (READ ONLY)

当表空间的状态为 READ ONLY 时,虽然可以访问表空间中的数据,但访问仅仅限于阅读,而不能进行任何更新或删除操作,目的是为了保证表空间的数据安全。

如果要将在表空间修改为 READ ONLY 状态,可以使用如下语法的 ALTER TABLESPACE 语句。

```
ALTER TABLESPACE tablespace_name READ ONLY;
```

将表空间的状态修改为 READ ONLY 之前,需要注意如下事项。

- ☐ 表空间必须处于 ONLINE 状态。
- ☐ 表空间不能包含任何事务的还原段。
- ☐ 表空间不能正处于在线数据库备份期间。

4. 读写 (READ WRITE)

当表空间的状态为 READ WRITE 时,可以对表空间进行正常访问,包括对表空间中的数据进

行查询、更新和删除等操作。

如果要将表空间修改为 READ WRITE 状态,可以使用如下语法的 ALTER TABLESPACE 语句。

```
ALTER TABLESPACE tablespace_name READ WRITE;
```

修改表空间的状态为 READ WRITE,也需要保证表空间处于 ONLINE 状态。

注意

无法将 Oracle 系统自定义的 system、temp 等表空间的状态设置为 OFFLINE 或 READ ONLY (除了 users 表空间以外)。

【练习 2】

假设要将 myspace 表空间的状态设置为 READ ONLY,修改后通过数据字典 dba_tablespaces 查看当前数据库中表空间的状态。语句如下所示。

```
SQL> ALTER TABLESPACE myspace READ ONLY;
```

表空间已更改。

```
SQL> SELECT tablespace_name , status FROM dba_tablespaces;
```

TABLESPACE_NAME	STATUS
SYSTEM	ONLINE
SYSAUX	ONLINE
UNDOTBS1	ONLINE
TEMP	ONLINE
USERS	ONLINE
MYSAPCE	READ ONLY

已选择 6 行。

其中,STATUS 列的值可能是 ONLINE、OFFLINE 或者 READ ONLY,分别对应在线且处于读写状态、离线状态、在线且处于只读状态。

5.2.3 创建表空间

Oracle 在创建表空间时将完成两个工作,一方面是在数据字典和控件文件中记录新建的表空间信息。另一方面是在操作系统中创建指定大小的操作系统文件,并作为与表空间对应的数据文件。

创建表空间需要使用 CREATE TABLESPACE 语句,其基本语法如下。

```
CREATE [ TEMPORARY | UNDO ] TABLESPACE tablespace_name
[
    DATAFILE | TEMPFILE 'file_name' SIZE size K | M [ REUSE ]
    [
        AUTOEXTEND OFF | ON
        [ NEXT number K | M MAXSIZE UNLIMITED | number K | M ]
    ]
    [ , ... ]
]
[ MINIMUM EXTENT number K | M ]
```

```

[ BLOCKSIZE number K]
[ ONLINE | OFFLINE ]
[ LOGGING | NOLOGGING ]
[ FORCE LOGGING ]
[ DEFAULT STORAGE storage ]
[ COMPRESS | NOCOMPRESS ]
[ PERMANENT | TEMPORARY ]
[
    EXTENT MANAGEMENT DICTIONARY | LOCAL
    [ AUTOALLOCATE | UNIFORM SIZE number K | M ]
]
[ SEGMENT SPACE MANAGEMENT AUTO | MANUAL ];

```

语法中各个参数的说明如下。

- ❑ **TEMPORARY | UNDO** 指定表空间的类型。TEMPORARY 表示创建临时表空间；UNDO 表示创建还原表空间；不指定类型，则表示创建的表空间为永久性表空间。
- ❑ **tablespace_name** 指定新表空间的名称。
- ❑ **DATAFILE | TEMPFILE 'file_name'** 指定与表空间相关联的数据文件。一般使用 DATAFILE，如果是创建临时表空间，则需要使用 TEMPFILE；file_name 指定文件名与路径。可以为一个表空间指定多个数据文件。
- ❑ **SIZE size** 指定数据文件的大小。
- ❑ **REUSE** 如果指定的数据文件已经存在，则使用 REUSE 关键字可以清除并重新创建该数据文件。如果文件已存在，但是没有指定 REUSE 关键字，则创建表空间时会报错。
- ❑ **AUTOEXTEND OFF | ON** 指定数据文件是否自动扩展。OFF 表示不自动扩展；ON 表示自动扩展。默认情况下为 OFF。
- ❑ **NEXT number** 如果指定数据文件为自动扩展，则 NEXT 子句用于指定数据文件每次扩展的大小。
- ❑ **MAXSIZE UNLIMITED | number** 如果指定数据文件为自动扩展，则 MAXSIZE 子句用于指定数据文件的最大大小。如果指定 UNLIMITED，则表示大小无限制，默认为此选项。
- ❑ **MINIMUM EXTENT number** 指定表空间中的盘区可以分配到的最小的尺寸。
- ❑ **BLOCKSIZE number** 如果创建的表空间需要另外设置其数据块大小，而不是采用初始化参数 db_block_size 指定的数据块大小，则可以使用此子句进行设置。此子句仅适用于永久性表空间。
- ❑ **ONLINE | OFFLINE** 指定表空间的状态为在线（ONLINE）或离线（OFFLINE）。如果为 ONLINE，则表空间可以使用；如果为 OFFLINE，则表空间不可使用。默认为 ONLINE。
- ❑ **LOGGING | NOLOGGING** 指定存储在表空间中的数据库对象的任何操作是否产生日志。LOGGING 表示产生；NOLOGGING 表示不产生。默认为 LOGGING。
- ❑ **FORCE LOGGING** 此选项用于强制表空间中的数据库对象的任何操作都产生日志，将忽略 LOGGING 或 NOLOGGING 子句。
- ❑ **DEFAULT STORAGE storage** 指定保存在表空间中的数据库对象的默认存储参数。当然数据库对象也可以指定自己的存储参数。

提示

BLOCKSIZE number 所设置的存储参数仅适用于数据字典管理的表空间。Oracle 的管理形式主要分为数据字典管理形式与本地化管理形式。不过，Oracle 11g 已经不再支持数据字典的管理形式。

- ❑ **COMPRESS | NOCOMPRESS** 指定是否压缩数据段中的数据。COMPRESS 表示压缩；NOCOMPRESS 表示不压缩。数据压缩发生在数据块层次中，以便压缩数据块内的行，消除列中的重复值。默认为 COMPRESS。

提示

对数据段中的数据进行压缩后，在检索数据时，Oracle 会自动对数据进行解压缩。这个过程不会影响数据的检索，但是会影响数据的更新和删除。

- ❑ **PERMANENT | TEMPORARY** 指定表空间中数据对象的保存形式。PERMANENT 表示持久保存；TEMPORARY 表示临时保存。
- ❑ **EXTENT MANAGEMENT DICTIONARY | LOCAL** 指定表空间的管理方式。DICTIONARY 表示采用数据字典的形式管理；LOCAL 表示采用本地化管理形式管理。默认为 LOCAL。
- ❑ **AUTOALLOCATE | UNIFORM SIZE number** 指定表空间中的盘区大小。AUTOALLOCATE 表示盘区大小由 Oracle 自动分配，此时不能指定大小；UNIFORM SIZE number 表示表空间中的所有盘区大小相同，都为指定值。默认为 AUTOALLOCATE。
- ❑ **SEGMENT SPACE MANAGEMENT AUTO | MANUAL** 指定表空间中段的管理方式。AUTO 表示自动管理方式；MANUAL 表示手动管理方式。默认为 AUTO。

【练习 3】

创建一个名称为 orclspace 的表空间，并设置表空间使用数据文件的初始大小为 20MB，每次自动增长 5MB，最大大小为 100MB，语句如下所示。

```
SQL> CREATE TABLESPACE orclspace
2  DATAFILE 'D:\oracle\files\orclspace.dbf'
3  SIZE 20M
4  AUTOEXTEND ON NEXT 5M
5  MAXSIZE 100M;
```

表空间已创建。

上述语句在创建 orclspace 表空间时忽略了许多属性的设置，也就是采用了许多默认设置。

提示

如果为数据文件设置了自动扩展属性，则最好同时为该文件设置最大大小限制。否则，数据文件的体积将会无限增大。

【练习 4】

创建 orclspace 表空间之后，下面通过数据字典 dba_tablespaces 查看该表空间的属性。语句如下所示。

```
SQL> select tablespace_name, logging, allocation_type, extent_management, segment_space_management
2  from dba_tablespaces
3  where tablespace_name = 'ORCLSPACE';
```

TABSPACE_NAME	LOGGING	ALLOCATION	EXTENT...	SEGMENT_SPACE...
-----	-----	-----	-----	-----

ORCLSPACE

LOGGING

SYSTEM

LOCAL

AUTO

下面对使用的 `dba_tablespaces` 数据字典的字段进行简单说明。

- ❑ **logging** 字段 表示是否为表空间创建日志记录。
- ❑ **allocation_type** 字段 表示表空间的盘区大小的分配方式。如果字段值为 `system`，则表示由 Oracle 系统自动分配，即为 `AUTOALLOCATE`。
- ❑ **extent_management** 字段 表示表空间盘区的管理方式。
- ❑ **segment_space_management** 字段 表示表空间中段的管理方式。

5.3

管理表空间

在实际环境中，一个 Oracle 数据库往往存在大量的系统表空间 and 用户自定义的表空间，而且随着时间的增长有些表空间可能不再需要或者损坏，这就需要数据库管理员对表空间进行管理。本节首先介绍 Oracle 提供的表空间管理方式，然后介绍各种管理操作的实现方法。

5.3.1 表空间的管理方式

根据表空间对区段管理方式的不同，表空间有两种管理方式分别是：数据字典管理的表空间和本地化管理的表空间。

由于使用数据字典管理表空间时，会遇到存在存储效率低、存储参数难以管理，以及磁盘碎片等问题，因此该方式已经被淘汰。Oracle 11g 中默认表空间都采用本地化管理方式，下面也仅介绍本地化管理表空间的方式。

本地化管理的表空间之所以能提高存储效率，其原因主要有以下几个方面。

- ❑ 采用位图的方式查询空闲的表空间、处理表空间中的数据块，从而避免使用 SQL 语句造成系统性能下降。
- ❑ 系统通过位图的方式，将相邻的空闲空间作为一个大的空间块，实现自动合并磁盘碎片。
- ❑ 区的大小可以设置为相同，即使产生了磁盘碎片，由于碎片是均匀统一的，也可以被其他实体重新使用。

如果需要查看表空间的管理方式，可以通过数据字典视图 `DBA_TABLESPACES` 中的 `EXTENT_MANAGEMENT` 字段来查询，如下所示。

```
SQL> SELECT TABLESPACE_NAME,EXTENT_MANAGEMENT
       2 FROM DBA_TABLESPACES;
```

TABLESPACE_NAME	EXTENT_MANAGEMENT
-----	-----
SYSTEM	LOCAL
SYSAUX	LOCAL
UNDOTBS1	LOCAL
TEMP	LOCAL
USERS	LOCAL
EXAMPLE	LOCAL
STUDENT	LOCAL
MYSpace	LOCAL

TEMTEST	LOCAL
TEMTESTGROUP001	LOCAL
BIGFILESAPCE	LOCAL
TABLESPACE_NAME	EXTENT_MANAGEMENT

5.3.2 增加数据文件

向表空间中增加数据文件需要使用 ALTER TABLESPACE 语句, 并指定 ADD DATAFILE 子句, 语法如下所示。

```
ALTER TABLESPACE tablespace_name
ADD DATAFILE
file_name SIZE number K | M
[
    AUTOEXTEND OFF | ON
    [ NEXT number K | M MAXSIZE UNLIMITED | number K | M ]
]
[ , ...];
```

【练习 5】

例如, 对练习 3 创建的 orclspace 表空间增加两个新的数据文件, 语句如下所示。

```
SQL> ALTER TABLESPACE orclspace
2 ADD DATAFILE
3 'D:\oracle\files\orclspace1.dbf'
4 SIZE 10M
5 AUTOEXTEND ON NEXT 5M MAXSIZE 40M ,
6 'D:\oracle\files\orclspace2.dbf'
7 SIZE 10M
8 AUTOEXTEND ON NEXT 5M MAXSIZE 40M ;
```

表空间已更改。

上述语句为 orclspace 表空间在 D:\oracle\files 目录下增加了名称为 orclspace1.dbf 和 orclspace2.dbf 的两个数据文件。

5.3.3 修改数据文件

修改表空间就是对表空间中数据文件的参数进行修改, 包括修改数据文件的大小、修改数据文件的自动扩展性以及数据文件的状态。

1. 修改表空间中数据文件的大小

如果表空间所对应的数据文件都被写满, 则无法再向该表空间中添加数据。这时, 可以通过修改表空间中数据文件的大小来增加表空间的大小。在修改之前可以通过数据库 dba_free_space 和数据字典 dba_data_files 查看表空间和数据文件的空间和大小信息。

修改数据文件需要使用 ALTER DATABASE 语句, 语法如下所示。

```
ALTER DATABASE DATAFILE file_name RESIZE newsize K | M;
```

语法说明如下所示。

□ **file_name** 数据文件的名称与路径。

❑ **RESIZE newsize** 修改数据文件的大小为 newsize。

【练习 6】

假设要修改 orclspace 表空间中数据文件的大小，语句如下所示。

```
SQL> ALTER DATABASE
      2 DATAFILE 'D:\oracle\files\orclspace1.dbf'
      3 RESIZE 10M;
```

数据库已更改。

上述语句将数据文件 D:\oracle\files\orclspace1.dbf 的大小修改为 10MB。

2. 修改表空间中数据文件的自动扩展性

将表空间的数据文件设置为自动扩展，目的是为了在表空间被填满后，Oracle 能自动为表空间扩展存储空间，而不需要管理员手动修改。

数据文件的扩展性除了在添加时指定外，也可以使用 ALTER DATABASE 语句修改其自动扩展性，语法如下所示。

```
ALTER DATABASE
DATAFILE file_name
AUTOEXTEND OFF | ON
      [ NEXT number K | M MAXSIZE UNLIMITED | number K | M ]
```

【练习 7】

假设要禁用 orclspace 表空间中 orclspace1.dbf 数据文件的自动扩展，语句如下所示。

```
SQL> ALTER DATABASE
      2 DATAFILE 'D:\oracle\files\orclspace1.dbf'
      3 AUTOEXTEND OFF;
```

3. 修改表空间中数据文件的状态

设置数据文件状态的语法如下所示。

```
ALTER DATABASE
DATAFILE file_name ONLINE | OFFLINE | OFFLINE DROP
```

其中，**ONLINE** 表示联机状态，此时数据文件可以使用；**OFFLINE** 表示脱机状态，此时数据文件不可使用，用于数据库运行在归档模式下的情况；**OFFLINE DROP** 与 **OFFLINE** 一样用于设置数据文件不可用，但它用于数据库运行在非归档模式下的情况。



如果将数据文件切换到 OFFLINE DROP 状态，则不能直接将其重新切换回 ONLINE 状态。

【练习 8】

假设要将 orclspace 表空间的 orclspace1.dbf 文件设置为 OFFLINE DROP 状态，如下所示。

```
SQL> ALTER DATABASE
      2 DATAFILE 'D:\oracle\files\orclspace1.dbf'
      3 OFFLINE DROP;
```

数据库已更改。

5.3.4 移动数据文件

数据文件是存储于磁盘中的物理文件，它的大小受到磁盘大小的限制。如果数据文件所在的磁盘空间不够，则需要将该文件移动到新的磁盘中保存。

【练习 9】

假设要移动 `orclspace` 表空间中的数据文件 `orclspace1.dbf`。具体步骤如下所示。

(1) 修改 `orclspace` 表空间的状态为 `OFFLINE`，如下所示。

```
SQL> ALTER TABLESPACE orclspace OFFLINE;
```

(2) 在操作系统中将磁盘中的 `orclspace1.dbf` 文件移动到新的目录中，例如移动到 `E:\oraclefile` 目录中。文件的名称也可以修改，例如修改为 `myoraclespace.dbf`。

(3) 使用 `ALTER TABLESPACE` 语句，将 `orclspace` 表空间中 `orclspace1.dbf` 文件的原名称与路径修改为新名称与路径，语句如下所示。

```
SQL> ALTER TABLESPACE orclspace  
2  RENAME DATAFILE 'D:\oracle\files\orclspace1.dbf'  
3  TO  
4  'E:\oraclefile\myoraclespace.dbf';
```

(4) 修改 `orclspace` 表空间的状态为 `ONLINE`，如下所示。

```
SQL> ALTER TABLESPACE orclspace ONLINE;
```

检查文件是否移动成功，也就是检查 `orclspace` 表空间的数据文件中是否包含了新的数据文件。使用数据字典 `dba_data_files` 查询 `orclspace` 表空间的数据文件信息，如下所示。

```
SQL> SELECT tablespace_name , file_name  
2  FROM dba_data_files  
3  WHERE tablespace_name = 'MYSPACE';
```

5.3.5 删除表空间

当不再需要某个表空间时可以删除该表空间，这要求用户具有 `DROP TABLESPACE` 系统权限。删除表空间需要使用 `DROP TABLESPACE` 语句，语法如下所示。

```
DROP TABLESPACE tablespace_name  
[ INCLUDING CONTENTS [ AND DATAFILES ] ]
```

语法说明如下所示。

- ❑ **INCLUDING CONTENTS** 表示删除表空间的同时，删除表空间中的所有数据库对象。如果表空间中有数据库对象，则必须使用此选项。
- ❑ **AND DATAFILES** 表示删除表空间的同时，删除表空间所对应的数据文件。如果不使用此选项，则删除表空间实际上仅是从数据字典和控制文件中将该表空间的有关信息删除，而不会删除操作系统中与该表空间对应的数据文件。

【练习 10】

假设要删除 `orclspace` 表空间，并同时删除该表空间中的所有数据库对象，以及操作系统中与之相对应的数据文件，语句如下所示。

```
SQL> DROP TABLESPACE orclspace
```

```
2 INCLUDING CONTENTS AND DATAFILES;
```

表空间已删除。

5.3.6 设置默认表空间

在新建用户时 Oracle 会为用户默认分配永久性表空间 **users**，以及临时表空间 **temp**。如果所有用户都使用默认的表空间，会增加 **users** 与 **temp** 表空间的负载压力和响应速度。

这时就可以修改用户的默认永久表空间和临时表空间。修改之前可以通过数据字典 **database_properties** 查看当前用户所使用的永久性表空间与临时表空间的名称，如下所示。

```
SQL> SELECT property_name , property_value , description
2 FROM database_properties
3 WHERE property_name
4 IN ('DEFAULT_PERMANENT_TABLESPACE' , 'DEFAULT_TEMP_TABLESPACE');
```

PROPERTY_NAME	PROPERTY_VALUE	DESCRIPTION
DEFAULT_TEMP_TABLESPACE	TEMP	Name of default temporary tablespace
DEFAULT_PERMANENT_TABLESPACE	USERS	Name of default permanent tablespace

其中，**default_permanent_tablespace** 表示默认永久性表空间；**default_temp_tablespace** 表示默认临时表空间。它们的值即为对应的表空间名。

Oracle 允许使用非 **users** 表空间作为默认永久性表空间，使用非 **temp** 表空间作为默认临时表空间。设置默认表空间需要使用 **ALTER DATABASE** 语句，语法如下所示。

```
ALTER DATABASE DEFAULT [ TEMPORARY ] TABLESPACE tablespace_name;
```

如果使用 **TEMPORARY** 关键字，则表示设置默认临时表空间；如果不使用该关键字，则表示设置默认永久性表空间。

【练习 11】

假设要将 **myspace** 表空间设置为默认永久性表空间，将 **mytemp** 表空间设置为默认临时表空间，语句如下所示。

```
SQL> ALTER DATABASE DEFAULT TABLESPACE myspace;

数据库已更改。

SQL> ALTER DATABASE DEFAULT TEMPORARY TABLESPACE mytemp;

数据库已更改。
```

使用数据字典 **database_properties** 检查默认表空间是否设置成功，语句如下所示。

```
SQL> SELECT property_name , property_value , description
2 FROM database_properties
3 WHERE property_name
4 IN ('DEFAULT_PERMANENT_TABLESPACE' , 'DEFAULT_TEMP_TABLESPACE');
```

PROPERTY_NAME	PROPERTY_VALUE	DESCRIPTION
DEFAULT_TEMP_TABLESPACE	TEMP	Name of default temporary tablespace
DEFAULT_PERMANENT_TABLESPACE	USERS	Name of default permanent tablespace

DEFAULT_TEMP_TABLESPACE	MYTEMP	Name of default temporary tablespace
DEFAULT_PERMANENT_TABLESPACE	MYSpace	Name of default permanent tablespace

提示

为数据库实例指定一个默认临时表空间组，其形式与指定默认临时表空间类似，只不过是使用临时表空间组名代替临时表空间名。例如为数据库实例指定临时表空间组为 group1，语句为：ALTER DATABASE DEFAULT TEMPORARY TABLESPACE group1;。

5.4

大文件表空间

大文件表空间是在 Oracle 10g 中提出来的，它只能对应惟一一个数据文件或临时文件，而不是像普通表空间由多个文件组成。

虽然大文件表空间只能对应一个数据文件或临时文件，但其对应的文件可达 4GB 个数据块大小。而普通表空间对应的文件最大可达 4MB 个数据块大小。

提示

如果数据块的大小被设置为 8KB，则大文件表空间对应的文件最大可以为 32TB；如果数据块的大小被设置为 32KB，则大文件表空间对应的文件最大可为 128TB。1TB = 1024GB。

5.4.1 创建大文件表空间

创建大文件表空间有三种方法，无论哪种方法都需要使用 BIGFILE 关键字，下面分别介绍。

1. 方法 1

在创建数据库时定义大文件表空间，并把它作为默认表空间，语句如下所示。

```
SQL> CREATE DATABASE
  2  SET DEFAULT BIGFILE TABLESPACE big_space_name
  3  DATAFILE 'D:\Oracle\Files\mybigspace.dbf'
  4  SIZE 1G;
```

一旦使用这种方法指定默认表空间为大文件表空间类型，以后创建的表空间都为大文件表空间，否则需要手动修改这个默认设置。

2. 方法 2

如果数据库已经创建，则可以使用 CREATE BIGFILE TABLESPACE 语句创建大文件表空间。例如下面的示例语句如下所示。

【练习 12】

例如要创建一个名称为 testbigspace 的大文件表空间可用如下语句。

```
SQL> CREATE BIGFILE TABLESPACE testbigspace
  2  DATAFILE 'D:\Oracle\Files\mybigspace.dbf'
  3  SIZE 1G;
```

创建成功之后可以通过数据字典 dba_data_files 验证表空间的大小信息，语句如下所示。

```
SQL> select tablespace_name, file_name, bytes/(1024*1024*1024) G
  2  from dba_data_files;
```

TABSPACE_NAME	FILE_NAME	G
-----	-----	-----
USERS	E:\APP\ADMINISTRATOR\ORADATA\ORCL\USERS01.DBF	0.00488281
UNDOTBS1	E:\APP\ADMINISTRATOR\ORADATA\ORCL\UNDOTBS01.DBF	0.29296875
SYS_AUX	E:\APP\ADMINISTRATOR\ORADATA\ORCL\SYS_AUX01.DBF	0.77709960
SYSTEM	E:\APP\ADMINISTRATOR\ORADATA\ORCL\SYSTEM01.DBF	0.68359375
EXAMPLE	E:\APP\ADMINISTRATOR\ORADATA\ORCL\EXAMPLE01.DBF	0.09765625
TESTBIGSPACE	D:\ORACLE\FILES\MYBIGSPACE.DBF	1

从结果可以看到大文件表空间 **testbigspace** 创建成功,其对应的 MYBIGSPACE.DBF 文件大小为 1GB。

3. 方法3

通过改变默认表空间为大文件表空间,可以使再创建的表空间都为大文件表空间。

【练习 13】

假设要修改数据库的默认表空间类型为大文件表空间 **testbigspace**, 语句如下所示。

```
SQL>alter tablespace set default bigfile testbigspace;
```

5.4.2 修改大文件表空间

在大文件表空间创建之后,可以根据需要对其进行修改,修改有两种方法,下面分别介绍。

第一种方法是在 ALTER TABLESPACE 语句中通过 RESIZE 子句来实现。

【练习 14】

例如要修改大文件表空间 **testbigspace** 中文件的大小为 4GB, 语句如下所示。

```
SQL>alter tablespace testbigspace resize 4G;
```

第二种方法是在 ALTER TABLESPACE 语句中通过 AUTOEXTEND ON 子句设置大文件表空间的文件为自动扩展。

【练习 15】

例如要为大文件表空间 **testbigspace** 开启自动扩展特性, 语句如下所示。

```
SQL>alter tablespace testbigspace autoextend on next 1G;
```

修改之后使用 **dba_data_files** 数据字典查看该文件的特性, 语句如下所示。

```
SQL> select tablespace_name,file_name,bytes/(1024*1024*1024) G,autoextensible
2   from dba_data_files
3  where tablespace_name='TESTBIGSPACE';
```

TABSPACE_NAME	FILE_NAME	G	AUTOEXTENSIBLE
-----	-----	----	-----
TESTBIGSPACE	D:\ORACLE\FILES\MYBIGSPACE.DBF	1	YES

从结果中可以看到大文件表空间 **TESTBIGSPACE** 的大小为 1GB, AUTOEXTENSIBLE 列为 YES 说明空间不足时会自动扩展。

【练习 16】

如果不知道当前使用的默认表空间类型可以使用如下语句查看。

```
SQL> select * from database_properties
```

```
2 where property_name='DEFAULT_TBS_TYPE';
```

PROPERTY_NAME	PROPERTY_VALUE	DESCRIPTION
-----	-----	-----
DEFAULT_TBS_TYPE	SMALLFILE	Default tablespace type

从结果的 PROPERTY_VALUE 列值可以看到，当前默认表空间类型 SMALLFILE，即标准小文件表空间。

提示

大文件表空间的删除方法与普通表空间一样。

5.5

临时表空间

在 Oracle 数据库中临时表空间适用于特定的会话活动，例如用户会话中的排序操作。排序的中间结果需要存储在某个区域，这个区域就是临时表空间。临时表空间的排序段是在实例启动后第一个排序操作时创建的。

默认情况下，所有用户都使用 temp 作为临时表空间。但是也允许使用其他表空间作为临时表空间，需要在创建用户时进行指定。

5.5.1 创建临时表空间

临时表空间是使用当前数据库的多个用户共享使用的，临时表空间中的区段在需要时按照创建临时表空间时的参数或者管理方式进行扩展。

使用临时表空间需要注意以下事项。

- ❑ 临时表空间只能用于存储临时数据，不能存储永久性数据。如果在临时表空间中存储永久性数据，将会出现错误。
- ❑ 临时表空间中的文件为临时文件，所以数据字典 dba_data_files 不再记录有关临时文件的信息。如果想要查看临时表空间的信息，可以查询 dba_temp_files 数据字典。
- ❑ 临时表空间的盘区管理方式都是 UNIFORM，所以在创建临时表空间时，不能使用 AUTOALLOCATE 关键字指定盘区的管理方式。

创建临时表空间时需要使用 TEMPORARY 关键字，并且与临时表空间对应的是临时数据文件由 TEMPFILE 关键字指定，也就是说临时表空间中不再使用数据文件，而使用临时数据文件。

【练习 17】

创建一个名称为 tempspace 的临时表空间，并设置临时表空间使用临时数据文件的初始大小为 10MB，每次自动增长 2MB，最大大小为 20MB，语句如下所示。

```
SQL> CREATE TEMPORARY TABLESPACE tempspace
2  TEMPFILE 'D:\oracle\files\tempespace.dbf'
3  SIZE 10M
4  AUTOEXTEND ON NEXT 2M MAXSIZE 20M;
```

表空间已创建。

【练习 18】

通过数据字典 dba_temp_files 查看临时表空间 tempspace 的信息，如下所示。

```
SQL> SELECT TABLESPACE_NAME, FILE_NAME, BYTES
2 FROM DBA_TEMP_FILES
3 WHERE TABLESPACE_NAME='TEMPSPACE';
```

TABLESPACE_NAME	FILE_NAME	BYTES
TEMPSPACE	D:\ORACLE\FILES\TEMPSPACE.DBF	10485760

临时表空间中的临时数据文件也是.DBF 格式的数据文件，但是这个数据文件与普通表空间或者索引的数据文件有很大不同，主要体现在如下几个方面。

- ☐ 临时数据文件总是处于 NOLOGGING 模式，因为临时表空间中的数据都是中间数据，只是临时存放的。它们的变化不需要记录在日志文件中，因为这些变化本身也不需要恢复。
- ☐ 临时数据文件不能设置为只读（READ ONLY）状态。
- ☐ 临时数据文件不能重命名。
- ☐ 临时数据文件不能通过 ALTER DATABASE 语句创建。
- ☐ 数据库恢复时不需要临时数据文件。
- ☐ 使用 BACKUP CONTROLFILE 语句时并不产生任何关于临时数据文件的信息。
- ☐ 使用 CREATE CONTROLFILE 语句不能设置临时数据文件的任何信息。
- ☐ 在初始化参数文件中，有一个名为 SORT_AREA_SIZE 的参数，这是排序区的容量大小。为了优化临时表空间中排序操作的性能，最好设置 UNIFORM SIZE 为该参数的整数倍。

5.5.2 管理临时表空间

创建临时表空间后，需要对它进行各种管理，主要包含如下几点。

- ☐ 增加临时数据文件。
- ☐ 修改临时数据文件的大小。
- ☐ 修改临时数据文件的状态。
- ☐ 切换临时表空间。
- ☐ 删除临时表空间。

【练习 19】

如果需要增加临时数据文件可以使用 ADD TEMPFILE 子句。下面的示例是为临时表空间 tempespace 增加一个临时数据文件。

```
SQL> alter tablespace tempespace
2 add tempfile 'D:\oracle\files\tempfile01.dbf' size 10m;
表空间已更改。
```

【练习 20】

假设要修改临时表空间 tempespace 中 tempfile01.dbf 文件的大小为 20MB，语句如下所示。

```
SQL> alter database tempespace
2 'D:\oracle\files\tempfile01.dbf' resize 20m;
数据库已更改。
```

技巧

由于临时文件中只存储临时数据，并且在用户操作结束后，系统将删除临时文件中存储的数据，所以一般情况下，不需要修改临时表空间的大小。

【练习 21】

假设要修改临时表空间 `temp space` 中 `tempfile01.dbf` 文件的状态更改为 `ONLINE`，语句如下所示。

```
SQL> alter database temp space
      2 'D:\oracle\files\tempfile01.dbf' online;
数据库已更改。
```

【练习 22】

将当前 Oracle 使用的默认临时表空间切换为 `temp space`，语句如下所示。

```
SQL> alter database default temporary tablespace temp space;
```

【练习 23】

假设要删除 `temp space` 临时表空间，语句如下所示。

```
SQL> drop tablespace temp space;
```

在删除前必须确保当前的临时表空间不在使用状态。因此删除默认的临时表空间之前必须先创建一个临时表空间，并切换至新的临时表空间。

5.5.3 临时表空间组

Oracle 11g 引入临时表空间组来管理临时表空间，一个临时表空间组中可以包含一个或者多个临时表空间。临时表空间组具有如下特点。

- ❑ 一个临时表空间组必须由至少一个临时表空间组成，并且无明确的最大数量限制。
- ❑ 如果删除一个临时表空间组的所有成员，该组也自动被删除。
- ❑ 临时表空间的名字不能与临时表空间组的名字相同。
- ❑ 在给用户分配一个临时表空间时，可以使用临时表空间组的名字代替实际的临时表空间名；在给数据库分配默认临时表空间时，也可以使用临时表空间组的名字。

使用临时表空间组有如下优点。

- ❑ 由于 SQL 查询可以并发使用几个临时表空间进行排序操作，因此 SQL 查询很少会出现排序空间超出，避免临时表空间不足所引起的磁盘排序问题。
- ❑ 可以在数据库级指定多个默认临时表空间。
- ❑ 一个并行操作的并行服务器将有效地利用多个临时表空间。
- ❑ 一个用户在不同会话中可以同时使用多个临时表空间。

【练习 24】

创建临时表空间组需要使用 `GROUP` 关键字。例如，创建一个临时表空间组，如下所示。

```
SQL> create temporary tablespace tempgroup1
      2 tempfile 'D:\Oracle\files\tempgroup1.dbf' size 10m
      3 tablespace group testtempgroup;
表空间已创建。
```

创建临时表空间组后，可以进行以下几个操作。

(1) 使用 `DBA_TABLESPACE_GROUPS` 数据字典查询临时表空间组的信息，如下所示。

```
SQL> select * from dba_tablespace_groups;
GROUP_NAME          TABLESPACE_NAME
```

```
-----
```

TESTTEMPGROUP	TEMPGROUP1
---------------	------------

(2) 向临时表空间组 **testtempgroup** 中增加一个临时表空间 **tempgroup2**, 如下所示。

```
SQL> create temporary tablespace tempgroup2
  2 tempfile 'D:\Oracle\files\tempgroup2.dbf' size 10m
  3 tablespace group testtempgroup;
表空间已创建。
```

(3) 将一个临时表空间组设置为默认的临时表空间, 可以使用 **DEFAULT** 关键字, 如下所示。

```
SQL> alter database default temporary tablespace testtempgroup;
数据库已更改。
```

(4) 将一个已经存在的临时表空间 **ORCLSPACE**, 移动到一个临时表空间组 **testtempgroup** 中, 如下所示。

```
SQL> alter tablespace orclspace
  2 tablespace group testtempgroup;
表空间已更改。
```

执行移动操作后, 使用 **DBA_TABLESPACE_GROUPS** 数据字典查询移动结果, 如下所示。

```
SQL> select * from dba_tablespace_groups;
GROUP_NAME          TABLESPACE_NAME
-----
TESTTEMPGROUP       TEMPGROUP1
TESTTEMPGROUP       TEMPGROUP 2
TESTTEMPGROUP       ORCLSPACE
```

(5) 删除临时表空间组, 也就是删除组成临时表空间组的所有临时表空间。

例如, 删除表空间组 **TESTTEMPGROUP** 中的表空间文件 **ORCLSPACE**, 如下所示。

```
SQL> drop tablespace orclspace including contents and datafiles;
表空间已删除。
```

使用 **DROP TABLESPACE** 语句删除表空间组中的 **TEMPGROUP1** 和 **TEMPGROUP2**。然后查看数据字典 **DBA_TABLESPACE_GROUPS**, 如下所示。

```
SQL> select * from dba_tablespace_groups;
未选定行。
```

由于表空间组不存在任何成员, 表空间组也随之被 **Oracle** 系统清理。

5.6

还原表空间

还原表空间在 **Oracle** 中主要用于存放还原段。例如, 如果一个用户要修改某个列的值, 将值从 **1** 修改为 **4**, 在更改的过程中其他用户要查看该数据时, 应该是 **1**, 因为数据还没有提交。所以, 为了保证这种读取数据的一致性, **Oracle** 使用了还原段, 在还原段中存放更改前的数据。

5.6.1 创建还原表空间

在 Oracle 中可以使用 CREATE UNDO TABLESPACE 语句创建还原表空间。创建之前首先了解 Oracle 对还原表空间的以下几点限制：

- ❑ 还原表空间只能使用本地化管理表空间类型，即 EXTENT MANAGEMENT 子句只能指定 LOCAL（默认值）。
- ❑ 还原表空间的盘区管理方式只能使用 AUTOALLOCATE（默认值），即由 Oracle 系统自动分配盘区大小。
- ❑ 还原表空间段的管理方式只能为手动管理方式，即 SEGMENT SPACE MANAGEMENT 只能指定 MANUAL。如果是创建普通表空间，则此选项默认为 AUTO；而如果是创建还原表空间，则此选项默认为 MANUAL。

【练习 25】

CREATE UNDO TABLESPACE 语句语法跟其他表空间的创建类似。例如，如下语句创建一个名为 undospace 的还原表空间。

```
SQL> CREATE UNDO TABLESPACE undospace
      2 DATAFILE 'D:\oracle\files\undospace.dbf'
      3 SIZE 10M;
```

表空间已创建。

5.6.2 管理还原表空间

还原表空间的管理与其他表空间的管理一样，都涉及到修改其中的数据文件、切换表空间以及删除表空间等操作。

1. 修改还原表空间的数据文件

由于还原表空间主要由 Oracle 系统自动管理，所以对还原表空间的数据文件的修改也主要限于以下几种形式。

- ❑ 为还原表空间添加新的数据文件。
- ❑ 移动还原表空间的数据文件。
- ❑ 设置还原表空间数据文件的状态为 ONLINE 或 OFFLINE。

提示

以上几种修改同样通过 ALTER TABLESPACE 语句实现，与普通表空间的修改方式一样。

2. 切换还原表空间

一个数据库中可以有多个还原表空间，但数据库一次只能使用一个还原表空间。默认情况下，数据库使用的是系统自动创建的 undotbs1 还原表空间。如果要将数据库使用的还原表空间切换成其他表空间，需要使用 ALTER SYSTEM 语句修改参数 undo_tablespace 的值。切换还原表空间后，数据库中新事务的还原数据将保存在新的还原表空间中。

【练习 26】

使用 ALTER SYSTEM 语句将数据库所使用的还原表空间切换为 undospace，语句如下所示。

```
SQL> ALTER SYSTEM SET undo_tablespace = 'UNDOSPACE';
系统已更改。
```

使用 SHOW PARAMETER 语句查看 undo_tablespace 参数的值，检查还原表空间是否切换成功，语句如下所示。

```
SQL> SHOW PARAMETER undo_tablespace;
```

NAME	TYPE	VALUE
undo_tablespace	string	UNDOSPACE

注意

如果切换时指定的表空间不是一个还原表空间,或者该还原表空间正在被其他数据库实例使用,将切换失败。

3. 修改撤销记录的保留时间

在 Oracle 中还原表空间中还原记录的保留时间由 `undo_retention` 参数决定,默认为 900 秒。900 秒之后,还原记录将从还原表空间中清除,这样可以防止还原表空间的迅速膨胀。

【练习 27】

使用 `ALTER SYSTEM` 语句修改 `undo_retention` 参数的值设置为 1200,即还原数据保留 1200 秒,语句如下所示。

```
SQL> ALTER SYSTEM SET undo_retention = 1200;
```

系统已更改。

使用 `SHOW PARAMETER` 语句查看修改后的 `undo_retention` 参数值,如下所示。

```
SQL> SHOW PARAMETER undo_retention;
```

NAME	TYPE	VALUE
undo_retention	integer	1200

注意

`undo_retention` 参数的设置不仅只对当前使用的还原表空间有效,而是应用于数据库中所有的还原表空间。

4. 删除还原表空间

删除还原表空间同样需要使用 `DROP TABLESPACE` 语句,但删除的前提是该还原表空间此时没有被数据库使用。如果需要删除正在被使用的还原表空间,则应该先进行还原表空间的切换操作。

【练习 28】

将数据库所使用的还原表空间切换为 `undotbs1`,然后删除还原表空间 `undospace`,语句如下所示。

```
SQL> ALTER SYSTEM SET undo_tablespace = 'UNDOTBS1';
```

系统已更改。

```
SQL> DROP TABLESPACE undospace INCLUDING CONTENTS AND DATAFILES;
```

表空间已删除。

5.6.3 更改还原表空间的方式

Oracle 11g 支持两种管理还原表空间的方式:还原段撤销管理(Rollback Segments Undo, RSU)和自动撤销管理(System Managed Undo, SMU)。其中,还原段撤销管理是 Oracle 的传统管理方式,要求数据库管理员通过创建还原段为撤销操作提供存储空间,这种管理方式不仅麻烦

而且效率也低;自动撤销管理是 Oracle 在 Oracle 9i 之后引入的管理方式,使用这种方式将由 Oracle 系统自动管理还原表空间。

一个数据库实例只能采用一种撤销管理方式,该方式由 `undo_management` 参数决定,可以使用 `SHOW PARAMETER` 语句查看该参数的信息,如下所示。

```
SQL> SHOW PARAMETER undo_management;
```

NAME	TYPE	VALUE
-----	-----	-----
undo_management	string	AUTO

如果参数 `undo_management` 的值为 `AUTO`,则表示还原表空间的管理方式为自动撤销管理;如果为 `MANUAL`,则表示为还原段撤销管理。

1. 自动撤销管理

如果选择使用自动撤销管理方式,则应将参数 `undo_management` 的值设置为 `AUTO`,并且需要在数据库中创建一个还原表空间。默认情况下,Oracle 系统在安装时会自动创建一个还原表空间 `undotbs1`。系统当前所使用的还原表空间由参数 `undo_tablespace` 决定。除此之外,还可以设置还原表空间中撤销数据的保留时间,即用户事务结束后,在还原表空间中保留撤销记录的时间。保留时间由参数 `undo_retention` 决定,其参数值的单位为秒。

使用 `SHOW PARAMETER undo` 语句,可以查看当前数据库的还原表空间的设置,如下所示。

```
SQL> SHOW PARAMETER undo;
```

NAME	TYPE	VALUE
-----	-----	-----
undo_management	string	AUTO
undo_retention	integer	900
undo_tablespace	string	UNDOTBS1

提示

如果一个事务的撤销数据所需的存储空间大于还原表空间中的空闲空间,则系统会使用未到期的撤销空间,这会导致部分撤销数据被提前从还原表空间中清除。

2. 还原段撤销管理

如果选择使用还原段撤销管理方式,则应将参数 `undo_management` 的值设置为 `MANUAL`,并且需要设置下列参数。

- ☐ `rollback_segments` 设置数据库所使用的还原段名称。
- ☐ `transactions` 设置系统中的事务总数。
- ☐ `transactions_per_rollback_segment` 指定还原段可以服务的事务个数。
- ☐ `max_rollback_segments` 设置还原段的最大个数。

5.7 实例应用：使用 OEM 管理表空间

5.7.1 实例目标

表空间和数据文件是 Oracle 数据库中非常重要的两个概念。表空间是一个逻辑概念,它和段、

区段以及数据库块组成了数据库的逻辑结构；而数据文件和操作系统块组成了数据库的物理结构。将数据库分为逻辑结构和物理结构是 Oracle 为满足不同操作系统之间的跨平台和可移植性而设计的。

Oracle 11g 默认采用本地化方式管理表空间，在 5.2.3 小节中介绍了如何使用 CREATE TABLESPACE 语句创建表空间。除此之外，用户也可以使用 Oracle 的 OEM 工具，在图形界面下管理表空间，本次实例主要介绍这种方式。

5.7.2 技术分析

表空间的维护是数据库管理员的一项重要任务，包括大文件表空间、临时表空间、默认表空间、还原表空间以及修改表空间等。

表空间创建后，可以把表空间中的数据文件移动到其他磁盘，以减少数据文件所在磁盘的压力。另外，不需要表空间时可以采用不同的方式删除表空间，但是删除时要注意一旦删除不可恢复。

5.7.3 实现步骤

(1) 在浏览器地址栏中请求 <https://whm:1158/em> (其中，whm 为笔者的计算机名)，登录到 OEM 页面，选择“服务器”选项页面，如图 5-3 所示。



图 5-3 进入 OEM 页面

(2) 在“服务器”页面的“存储”一档中，单击【表空间】链接，进入表空间管理页面，在该页面中显示了已经存在的表空间信息，如图 5-4 所示。

(3) 单击图 5-4 所示页面中的【创建】按钮，即可进入创建表空间页面。在创建表空间页面中，在“名称”文本框中输入表空间名称，对其他内容可以使用默认值，如图 5-5 所示。

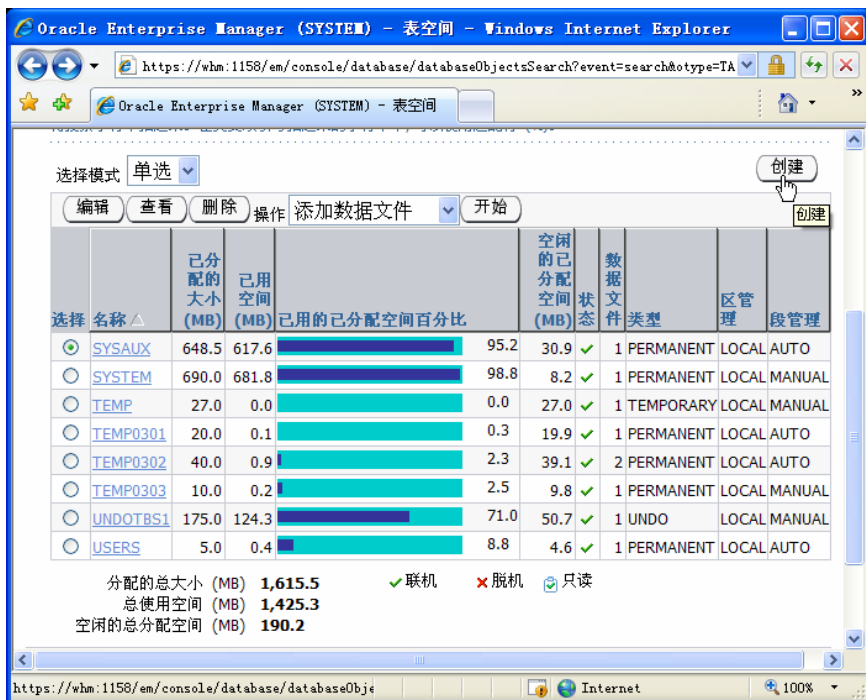


图 5-4 表空间管理页面



图 5-5 创建表空间

(4) 如图 5-5 所示, 页面的下半部分可以指定表空间的数据文件, 单击【添加】按钮, 进入到添加数据文件页面, 输入数据文件的名称、路径、文件大小和存储方式等, 如图 5-6 所示。



图 5-6 添加数据文件

(5) 单击【继续】按钮, 将返回到图 5-5 所示的创建表空间页面, 在数据文件一栏中将显示添加的数据文件, 如图 5-7 所示。

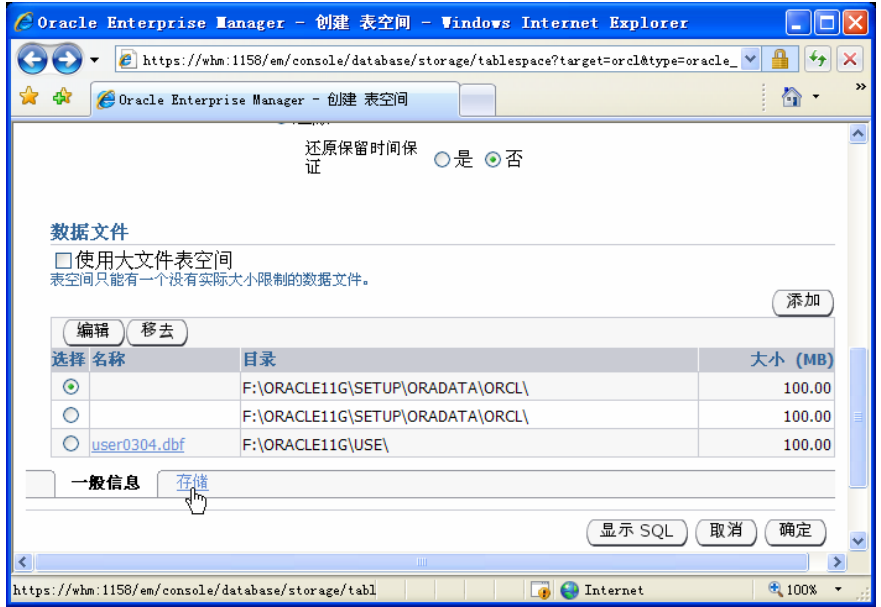


图 5-7 显示创建的数据文件

(6) 单击如图 5-7 中的【存储】链接, 将进入创建表空间的存储页面。在“存储”选项页面中可以设置表空间的区管理方式、段空间管理方式、是否压缩、以及是否启用事件记录等选项, 如图 5-8 所示。

(7) 如果需要查看所生成的 SQL 语句, 单击【显示 SQL】按钮即可进入显示 SQL 页面, 如图 5-9 所示。



图 5-8 表空间的存储选项

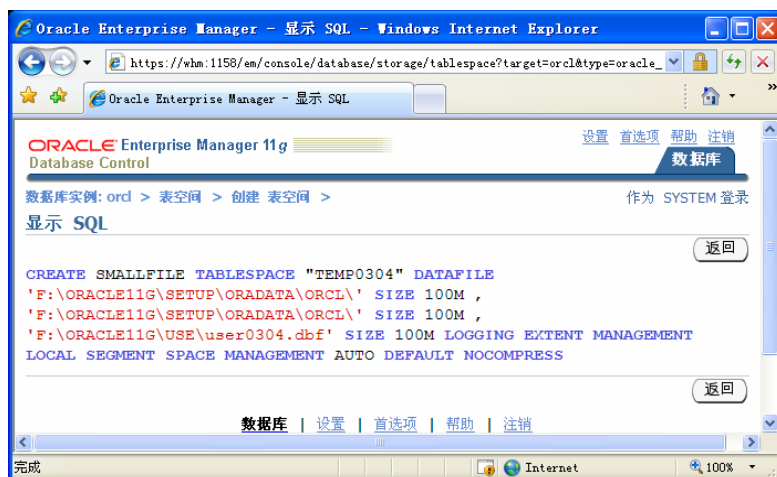


图 5-9 创建表空间的 SQL 语句

(8) 单击【返回】按钮，回到图 5-8 所示的页面单击【确定】按钮。这样就使用 OEM 创建了一个表空间 TEMP0304 和数据文件 use0304.dbf。

5.8

拓展训练

操作 Oracle 表空间

本课讨论了表空间的逻辑结构和物理结构之间的关系，然后讲解各种类型表空间的创建及管理操作。本次训练要求读者完成如下表空间的操作。

(1) 查看当前 Oracle 数据库都使用了哪些表空间。

(2) 创建一个名称为 `schoolspace` 的表空间，并设置表空间使用数据文件的初始大小为 10MB，每次自动增长 2MB，最大大小为 50MB。

(3) 向 `schoolspace` 表空间中添加一个名为 `schooldf2` 的数据文件。

(4) 设置第 2 步创建的数据文件为自动扩展。

(5) 将 `schoolspace` 设置为默认表空间。

(6) 创建一个名为 `schooltemp space` 的临时表空间。

(7) 将 `schooltemp space` 设置为默认临时表空间。

(8) 创建一个名为 `schoolundospace` 的还原表空间。

5.9

课后练习

一、填空题

1. Oracle 数据文件的逻辑结构中_____是表空间内的一个逻辑存储空间。
2. 在创建临时表空间时应使用_____关键字为其指定临时文件。
3. 创建撤销表空间需要使用_____关键字。
4. Oracle 中用户默认的永久性表空间为_____，默认的临时表空间为 `temp`。
5. 在空白填写合适语句，使其可以创建一个临时表空间 `temp`。

```
CREATE _____ TABLESPACE temp
_____ 'F:\oraclefile\temp.dbf'
SIZE 10M
AUTOEXTEND ON
NEXT 2M
MAXSIZE 20M;
```

6. Oracle 11g 管理还原表空间的方式有_____和自动撤销管理。

二、选择题

1. 下列不属于 Oracle 中数据文件逻辑结构组成部分的是_____。
 - A. 数据库块
 - B. 操作系统块
 - C. 数据块
 - D. 区段
2. 下面哪些不属于表空间的状态属性？_____
 - A. ONLINE
 - B. OFFLINE
 - C. OFFLINE DROP
 - D. READ ONLY
3. 将表空间的状态切换为 OFFLINE 时，不可以指定下面哪个参数？_____
 - A. TEMP
 - B. IMMEDIATE
 - C. NORMAL
 - D. FOR RECOVER

4. 假设要删除表空间 `space`，并同时删除其对应的数据文件，可以使用下列哪条语句？_____
- A. `DROP TABLESPACE space;`
 - B. `DROP TABLESPACE space AND DATAFILES;`
 - C. `DROP TABLESPACE space INCLUDING DATAFILES;`
 - D. `DROP TABLESPACE space INCLUDING CONTENTS AND DATAFILES;`
5. 下列将临时表空间 `temp` 设置为默认临时表空间的语句正确的是_____。
- A. `ALTER DATABASE DEFAULT TEMPORARY TABLESPACE temp;`
 - B. `ALTER DEFAULT TEMPORARY TABLESPACE TO temp;`
 - C. `ALTER DATABASE DEFAULT TABLESPACE temp;`
 - D. `ALTER DEFAULT TABLESPACE TO temp;`
6. 假设有一个临时表空间 `temp1` 存放在临时表空间组 `group1` 中，现在修改 `temp1` 表空间所在的组为 `group2`。下面对修改后的结果叙述正确的是_____。
- A. 由于数据库实例中并不存在 `group2` 组，所以上述操作将执行失败
 - B. 修改后 `temp1` 表空间将被删除
 - C. 修改后数据库实例中将存在两个临时表空间组 `group1` 和 `group2`
 - D. 修改后数据库实例中将只存在一个临时表空间组 `group2`

三、简答题

1. 分析表空间在逻辑结构中的位置及其作用。
2. 罗列表空间的状态值，它们分别表示什么意思？
3. 如何查看当前正在使用的表空间，切换表空间和设置默认表空间。
4. 简述创建大文件表空间的几种方法。
5. 创建临时表空间时需要注意哪些问题，创建方法是什么？
6. 在实际应用中，需要临时创建一个表来使用，那么是否可以将该表创建在临时表空间中？
7. Oracle 11g 支持哪些方式的还原表空间，它们之间有什么区别？