

北京林业大学

# 数据库原理与应用

# 创建与使用索引



## 索引概述

- ◆ 索引是一种可以加快检索的数据库结构。
- ◆ 通过创建良好的索引，可以显著提高数据库查询和应用程序的性能。
- ◆ 索引一旦创建，将由DBMS自动管理和维护。当操作数据时，DBMS会自动更新索引。
- ◆ 要避免在一个表中创建大量的索引，否则会影响数据操作的性能，降低系统的响应速度。





# 索引的类型

## 聚集索引

在聚集索引中，表中行的物理存储顺序与索引键的逻辑顺序相同。

由于真正的物理存储只有一个，因此，一个表只能包含一个聚集索引。



# 索引的类型

## 非聚集索引

非聚集索引与聚集索引具有相似的索引结构。

不同的是，非聚集索引不影响数据行的物理存储顺序。



## 索引的类型

### 唯一索引

唯一索引能够保证索引键中不包含重复的值，从而使表中的每一行在某种方式上具有唯一性。





## 索引的类型

### 视图索引

对视图而言，查询动态生成的结果集开销很大，特别是涉及对大量行进行复杂处理的视图。

如果在查询中频繁地引用这类视图，可通过对视图创建唯一聚集索引来提升性能，这类索引称为视图索引。



## 索引的类型

### 全文索引

全文索引是目前搜索引擎的关键技术之一。试想在1兆大小的文件中搜索一个词，可能需要几秒，在更大的文件中搜索开销会更大。

为加快此类检索速度，出现了全文索引技术，也称倒排文档技术。





## 索引的类型

### XML索引

可以对xml数据类型列创建XML索引。  
XML索引对列中xml实例的所有标记、  
值和路径进行索引，提高查询性能。



# 创建索引

## 用SQL语句创建索引的基本语法:

```
CREATE [UNIQUE] [CLUSTERED |  
NONCLUSTERED] INDEX index_name  
ON table_or_view_name (column_name [ASC |  
DESC] [...n])  
[WITH<index_option> [...n] ]  
[ON {filegroup_name | “default” } ]
```



# 创建索引

## 用SQL语句创建索引

**[例] 为表SC在SNo和CNo上建立惟一索引。**

```
CREATE UNIQUE INDEX SCI ON  
SC(SNo,CNo)
```

**[例] 为教师表T在TN上建立聚集索引。**

```
CREATE CLUSTER INDEX TI ON T(TN)
```





# 修改索引

## 修改索引的SQL命令语法如下:

```
ALTER INDEX { index_name | ALL }  
ON table_or_view_name  
{ REBUILD  
  [ [PARTITION = ALL]  
    [ WITH ( <rebuild_index_option> [ ,...n ] ) ]  
    | [ PARTITION = partition_number  
      [ WITH ( <single_partition_rebuild_index_option>  
        [ ,...n ] )  
      ] ] ]  
| DISABLE  
| REORGANIZE  
  [ PARTITION = partition_number ]  
  [ WITH ( LOB_COMPACTION = { ON | OFF } ) ]  
| SET ( <set_index_option> [ ,...n ] )  
}[ ; ]
```



# 修改索引

## 修改索引的参数介绍:

- (1) REBUILD: 删除索引并且重新生成索引。
- (2) PARTITION: 指定只重新生成或重新组织索引的一个分区。
- (3) DISABLE: 将索引标记为禁用, 从而不能由数据库引擎使用。
- (4) REORGANIZE: 重新组织索引。



# 删除索引

## 用SQL语句删除索引

`DROP INDEX <table or view name>.<index name>`

`DROP INDEX <index name> ON <table or view name>`





## 查看索引



用Sp\_helpindex存储过程查看索引

Sp\_helpindex [ @objname = ] 'name'



表的名称

[例] 查看表SC的索引。

EXEC Sp\_helpindex SC



## 创建与使用索引



如果要更改索引名称，可利用  
Sp\_rename 存储过程更改，其语法如下：

`Sp_rename '数据表名.原索引名', '新索引名'`

**[例]** 更改T表中的索引TI名称为T\_Index。

```
EXEC Sp_rename 'T.TI', 'T_Index'
```