



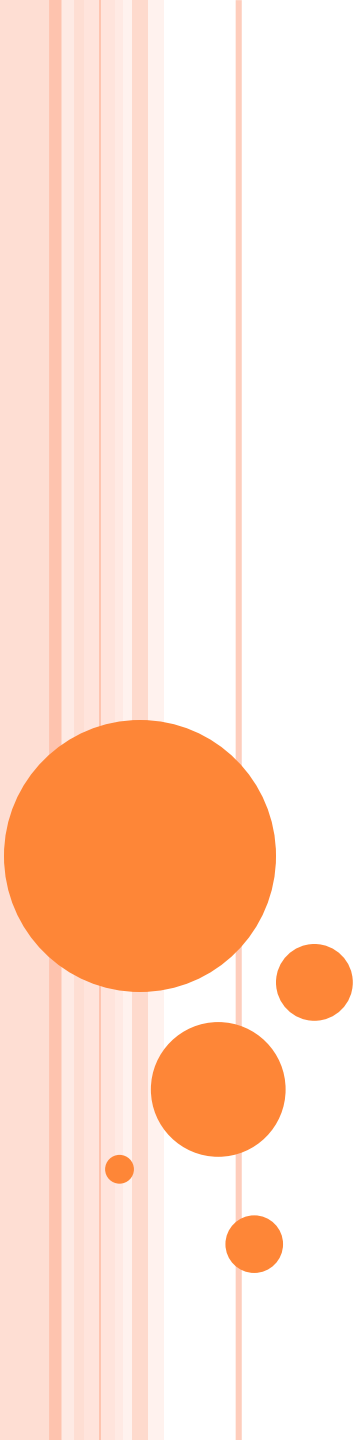
数据库系统

Database System

主讲：张仲楠 教授

Email: zhongnan_zhang@xmu.edu.cn

Office: 西部片区5#-202



实验六 数据库内模式

数据库内模式

- 分区表
- UNIQUE索引



分区表

- 分区表是把逻辑上的一张表根据某种方案分成几张物理块进行存储，这张逻辑上的表称之为分区表，物理块称之为分区。
- 分区表是一张逻辑表，不存储数据，数据实际是存储在分区上的。
- 普通表若要转成分区表，需要新建分区表，然后把普通表中的数据导入到新建的分区表中。

分区表

- 改善查询性能：对分区对象的查询可以仅搜索自己关心的分区，提高检索效率。
- 增强可用性：如果分区表的某个分区出现故障，表在其他分区的数据仍然可用。
- 方便维护：如果分区表的某个分区出现故障，需要修复数据，只修复该分区即可。
- 均衡I/O：可以把不同的分区映射到不同的磁盘以平衡I/O，改善整个系统性能。

分区表

- openGauss数据库支持这些划分类型：
- 范围分区表：指定一个或多个列划分为多个范围，每个范围创建一个分区，用来存储相应的数据。例如可以采用日期划分范围，将销售数据按照月份进行分区。
- 列表分区表：直接按照一个列上的值来划分出分区。例如可以采用销售门店划分销售数据。
- 哈希分区表：根据表的一列，通过内部哈希算法将要插入表的记录划分到对应的分区中。



范围分区

```
create table purchase_part  
(  
transaction_id int primary key,  
item_id int not null,  
item_description varchar(30),  
transaction_date date not null  
)  
partition by range (transaction_date)  
(  
partition part_01 values less than('2006-04-01'),  
partition part_02 values less than('2006-07-01'),  
partition part_03 values less than('2006-10-01'),  
partition part_04 values less than('2007-01-01')  
);
```



范围分区

- purchase_part
 - 列
 - 约束
 - 索引
 - 分区
 - part_01
 - part_02
 - part_03
 - part_04

一般 列 约束 索引 分区			
包含搜索内容 请输入搜索内容			
	分区名称	分区边界	表空间
1	part_01	2006-04-01	[NULL]
2	part_02	2006-07-01	[NULL]
3	part_03	2006-10-01	[NULL]
4	part_04	2007-01-01	[NULL]



问题1

- `insert into purchase_part values (1, 12, 'BOOKS', sysdate);`
- `insert into purchase_part values (3, 13, 'TABLES', '2006-05-30');`
- `insert into purchase_part values (4, 14, 'DESKS', '2006-02-23');`
- `insert into purchase_part values (5, 15, 'CHAIRS', '2006-10-26');`
- `insert into purchase_part values (6, 12, 'BOOKS', '2006-04-30');`

结果如何？




问题2

- (1) 列出分区part_02中的所有购买记录
- (2) 将分区part_01中的有关 'DESKS' 的 transaction_date改为2006-02-24, 查询分区part_01
- (3) 将分区part_01中的有关 'DESKS' 的 transaction_date改为2006-12-24, 查询分区part_01
- (4) 将购买记录(7, 12, 'BOOKS', '2006-09-09')插入 purchase_part
- (5) 分别列出分区part_03和part_04中的所有购买记录



列表分区

```
create table user_by_city(  
  userID int primary key,  
  username varchar(25),  
  city varchar(10)  
)  
partition by list(city) (  
  partition p1 values ('北京', '上海', '天津'),  
  partition p2 values ('深圳', '广州'),  
  partition p3 values ('郑州', '武汉', '西安'),  
  partition p4 values ('大连', '沈阳')  
);
```



问题3

(1) 插入元组 (1, 'zhangsan', '北京');

(2) 将用户1的城市改为'深圳' ;

(3) 插入元组 (2, 'lisi', '南京');

(4) 尝试增加新分区p5, 该分区的城市包括 ('成都', '重庆')



哈希分区

- 无法对各分区中可能具有的记录数目进行预测时，可创建哈希分区

```
create table itemInfo  
( item_id int,  
  description varchar(30))  
partition by hash(item_id)  
(partition p1,  
 partition p2);
```



问题4

(1) 插入如下元组:

(1, 'tv'), (2, 'bike'), (3, 'book'), (4, 'table')

(2) 列出分区 p1 中的所有元组

(3) 插入元组 (5, 'chair'), 看看它会出现哪个分区中



UNIQUE索引

- 问题5：当表中还没有数据时，对表中没有任何限制的某一列创建UNIQUE索引。之后再向表中插入数据，对以下几种情况进行试验：
 - 该列所对应的值没有重复值出现，结果如何
 - 该列所对应的值有重复值出现，结果如何
 - 第一次对该列插入NULL，结果如何
 - 第二次对该列插入NULL，结果如何
- 问题6：当表中已有如下情况的数据时，对该列创建UNIQUE索引：
 - 该列所对应的值没有重复值出现，结果如何
 - 该列所对应的值有重复值出现，结果如何
 - 该列所对应的值有一个NULL，结果如何
 - 该列所对应的值有多个NULL，结果如何



作业提交须知

- 需提交的文件包括：

实验报告.doc（包含sql代码文本和运行结果截图）

```
1  
select sname, cno, grade  
from student, sc  
where student.sno=sc.sno;
```



文本

	SNAME	CNO	GRADE
1	李勇	1	92
2	李勇	2	85
3	李勇	3	88
4	刘晨	2	90
5	刘晨	3	80



截图

- 在课程中心(1nt.xmu.edu.cn)提交实验报告
- 截止时间：2024-06-05 23:59:59



作业提交须知

○ 迟交作业

```
update sc
set grade=0
where sno='2432018220xxxx'
      and grade is null
      and miss_deadline = 1;
```

○ 抄袭作业

```
update sc
set grade=0
where sno='2432018220xxxx'
      and grade is null
      and plagiarize = 1;
```

