

数据库系统课程教案

第 21 单元

学时：2

教材内容	第 10 章 数据库恢复技术
基本知识点	事务的概念和 ACID 性质、恢复技术、故障的种类及恢复策略、检查点技术、Oracle 的恢复技术
教学重点	事务的基本概念和事务的 ACID 性质；数据库恢复的实现技术；日志文件的内容和作用；登记日记文件所要遵循的准则、恢复的基本原理，针对不同故障的恢复的策略和方法
教学难点	日志文件的使用，系统故障恢复策略
要求掌握内容	事务的基本概念和事务的 ACID 性质；数据库恢复的实现技术；日志文件的内容和作用；登记日记文件所要遵循的准则、恢复的基本原理，针对不同故障的恢复的策略和方法
教学思路，采用的教学方法和辅助手段，板书设计，重点如何突出，难点如何解决，师生互动等	<p>教学思路：</p> <p>一、复习旧课，巩固上次授课主要内容</p> <ol style="list-style-type: none"> 1、简述查询优化的概念。 2、掌握如何画出一个查询的语法树以及优化后的标准语法树。 <p>二、导入新课，明确本次授课的目的与要求</p> <ol style="list-style-type: none"> 1、掌握事务的概念和性质。 2、理解数据库故障的三种类型。 3、掌握数据库恢复的技术和策略。 4、理解数据库镜像的概念。 5、掌握 Oracle 的恢复技术。 <p>三、讲解本次授课的具体内容</p> <p>教学方法：整合教学内容，强化基础训练；努力营造生动活泼的课堂气氛，搭建师生间良好的沟通渠道；采用多媒体教学与传统的板书设计相结合的方式，教学手段灵活多变。</p> <p>辅助手段：通过 PPT 幻灯片演示结合板书设计和举例加以阐述。</p>
本章思考题和作业	P305 第 1、5、8、9 题
主要教材参考资料	<ol style="list-style-type: none"> 1. 《数据库系统概论》，萨师煊，王珊，高等教育出版社，2014.9 2. 《数据库系统概论学习指导与习题解答》，王珊，张俊，高等教育出版社，2015.7
备 注	

本次授课具体内容

第 10 章 数据库恢复技术

10.1 事务的基本概念

(-)数据库系统中的数据是由 DBMS 统一管理和控制的，为了适应数据共享的环境，DBMS 必须提供数据保护能力，以保证数据库中数据的安全可靠和正确有效。数据库保护包括数据库恢复、安全性、完整性和并发控制四大技术。

(-)事务(transaction)的概念：用户定义的一个数据库操作序列，这些操作要么全做，要么全不做，是一个不可分割的工作单位。在关系数据库中，一个事务可以是一条 SQL 语句，一组 SQL 语句或整个程序。事务不同于程序，一个程序通常包含多个事务。事务是数据库恢复和并发控制的基本单位。如从 A 帐户转帐到 B 帐户 100 元的过程是一个事务，该事务包括了两个更新操作： $A=A-100$ ， $B=B+100$ ，这两个操作不可分割：要么全做，要么全不做。事务的开始与结束可以由用户显式控制，如果没有显式地定义事务，则由 DBMS 按缺省规定自动地划分事务。显式定义为：

Begin Transction		Begin Transction
.....	或
Commit		RollBack

其中，Begin Transction 表示事务的开始，Commit 为事务提交，即告诉事务管理器事务中的所有操作都已完成，数据库处于另一个一致性状态；而 RollBack 为事务回滚，即告诉事务管理器事务执行时发生故障，所有已完成的操作必须全部撤销，滚回到事务开始的状态。这里的操作指对数据库的更新操作。

(-)事务的 ACID 性质

- 1、原子性(Atomicity)：事务是数据库的逻辑工作单位，事务包括的诸操作要么全做，要么全不做。
- 2、一致性(Consistency)：事务的执行必须保证数据库从一个一致性状态转到另一个一致性状态。
- 3、隔离性(Isolation)：一个事务的执行不能被其他事务干扰，并发执行的各个事务间应互相独立。
- 4、持久性(Durability)：事务一旦提交，它对数据库中数据的改变应是永久的。

10.2 数据库恢复概述

(-)由于计算机系统中硬件故障、软件错误、操作员失误以及恶意破坏是不可避免的，这些故障轻则造成运行事务非正常中断，影响数据库的正确性，重则破坏数据库，使数据库全部或部分数据丢失。DBMS 必须具有把数据库从错误状态恢复到某一已知的正确状态的功能，这就是数据库的恢复。

(-)恢复子系统是 DBMS 的一个重要组成部分，保证故障发生后能把数据库中的数据从错误状态恢复到某一已知的正确状态，保证事务 ACID。恢复技术是衡量系统优劣的重要指标。

10.3 故障的种类

(-)事务故障：某个事务在运行过程中由于种种原因未运行至正常结束点就夭折了，造成数据库可能处于不正确状态。事务故障的原因：违反了某些完整性限制、输入数据有误、运算溢出、并行事务发生死锁等。事务故障分为可预期的(应用程序可以发现的故障，如违反了某些完整性限制、输入数据有误等)和不可预期的(应用程序无法发现只能由系统处理的故障，如运算溢出、发生死锁等)两大类。

(二)系统故障(软故障)：指造成系统停止运转的任何事件，使得系统要重新启动。这类故障影响所有正在运行的事务，但不破坏整个数据库。这时内存数据丢失，所有运行事务都异常终止。系统故障的原因：特定类型的硬件错误(如 CPU 故障)、操作系统故障、DBMS 代码错误、突然停电等。

(三)介质故障(硬故障)：使存储在外存中的数据部分丢失或全部丢失。介质故障 发生的可能性较小，但破坏性大，破坏数据库或部分数据库，并影响正在存取这部分数据的所有事务。介质故障的原因：磁盘损坏、磁头碰撞、瞬时强磁场干扰等。

10.4 恢复的实现技术

(一)恢复机制涉及的两个关键问题：如何建立冗余数据(数据转储 backup、登录日志文件 logging)、如何利用这些冗余数据实施数据库恢复。

(二)数据转储：指 DBA 将整个数据库复制到磁带或另一磁盘上保存起来的过程。这些备用的数据文本称为后备副本或后援副本。

(三)登记日志文件(Logging)：日志文件(log)是用来记录事务对数据库更新操作的文件。

1、日志文件的作用：事务故障恢复和系统故障恢复必须用日志文件。

2、登记日志文件时必须遵循两条原则

(1)登记的次序严格按照并行事务执行的时间次序。

(2)必须先写日志文件，后写数据库写日志文件操作。

把对数据的修改写到数据库中和把表示这个修改的日志记录写到日志文件中是两个不同的操作。有可能在这两个操作之间发生故障，即两个写操作只完成了一个。若先写了数据库修改，而在日志文件中没有登记下这个修改，则以后就无法恢复这个修改了；若先写日志，但没有修改数据库，按日志文件恢复时只不过是多执行一次不必要的 UNDO 操作，并不会影响数据库的正确性。

10.5 恢复策略

(一)事务故障的恢复：事务故障指事务在运行至正常终止点前被中止，这时恢复子系统应利用日志文件撤消(UNDO)此事务已对数据库进行的修改。恢复由系统自动完成，对用户是透明的。恢复步骤是：

1、反向扫描文件日志(即从最后向前扫描日志文件)，查找该事务的更新操作。

2、对该事务的更新操作执行逆操作。即若记录中是插入操作，“更新前的值”为空，则相当于做删除操作；若记录中是删除操作，“更新后的值”为空，则相当于做插入操作；若记录中是修改操作，则用修改前值代替修改后值。

3、继续反向扫描日志文件，查找该事务的其他更新操作，并做同样处理。

4、如此处理下去，直至读到此事务的开始标记，事务故障恢复就完成了。

(二)系统故障的恢复：系统故障造成数据库不一致状态的原因有两个，一是未完成事务对数据库的更新已写入数据库，二是一些已提交事务对数据库的更新还留在缓冲区来不及写入数据库。因此恢复操作要 UNDO 故障发生时未完成的事务，REDO 已完成的事务。恢复由系统在重新启动时自动完成，不需要用户干预。恢复步骤是：

1、正向扫描日志文件(即从头扫描日志文件)，找出在故障发生前已经提交的事务(这些事务既有 Begin Transaction 记录，又有 Commit 记录)，将其事务标识放入 REDO 队列；同时找出故障发生时尚未完成的事务(这些事务只有 Begin Transaction 记录、但无 Commit 记录)，将其事务标识放入 UNDO 队列。

2、反向扫描日志文件，对 UNDO 队列中各个事务进行撤销处理。

3、正向扫描日志文件，对 REDO 队列中各个事务进行重做处理。

注意：必须先 UNDO 再 REDO。

(三)介质故障的恢复：发生介质故障后，磁盘上的物理数据和日志文件被破坏，这是最严重的一种故障。恢复方法是重装数据，然后重做已完成的事务。具体步骤是：

1、装入最新的数据库后备副本(离故障发生时刻最近的转储副本)，使数据库恢复到最近一次转储时的一致性状态。对于静态转储的数据库副本，装入后数据库即处于一致性状态；对于动态转储的数据库副本，还须同时装入转储时刻的日志文件副本，利用与恢复系统故障相同的方法(即 REDO+UNDO)，才能将数据库恢复到一致性状态。

2、装入相应的日志文件副本(转储结束时刻的日志文件副本)，重做已完成的事务。即首先扫描日志文件，找出故障发生时已提交的事务的标识，将其记入重做队列。然后正向扫描日志文件，对重做队列中的所有事务进行重做处理。即将日志记录中“更新后的值”写入数据库。

介质故障的恢复需要 DBA 介入，但 DBA 只需要重装最近转储的数据库副本和有关的各日志文件副本，然后执行系统提供的恢复命令即可，具体的恢复操作仍由 DBMS 完成。

10.6 具有检查点的恢复技术

(一)引入检查点的原因：利用日志文件进行数据库恢复时，恢复子系统必须搜索日志，确定哪些事务需要 REDO，哪些事务需要 UNDO。这样做会产生两个问题：一是搜索整个日志将耗费大量时间，二是很多需要 REDO 处理的事务实际上已经将其更新结果写入数据库，然而恢复子系统又重新执行了这些操作，浪费了大量时间。为此，引入了检查点技术。

(二)具有检查点的恢复技术：在日志文件中增加检查点 checkpoint 记录(记载了检查点时刻所有正在执行的事务清单和这些事务最近一个日志记录的地址)，增加一个重新开始文件(记载了各个检查点记录在日志中的地址)，并让恢复子系统在登录日志文件期间动态地维护日志。

(三)具有检查点技术的恢复步骤

1、从重新开始文件中找到最后一个检查点记录在日志中的地址，由该地址在日志文件中找到最后一个检查点记录。

2、由该检查点记录得到检查点建立时刻所有正在执行的事务清单 ACTIVE_LIST，建立 UNDO 队列和 REDO 队列，把 ACTIVE_LIST 暂时放入 UNDO 队列，REDO 队列暂为空。

3、从检查点开始正向扫描日志，如有新开始事务加入 UNDO 队列，如有提交的事务加入 REDO 队列。

4、对 UNDO 队列中的事务 UNDO 处理，对 REDO 队列中的事务 REDO 处理。

10.7 数据库镜像

(一)介质故障严重影响了数据库的可用性，而且恢复比较费时。为预防介质故障，DBA 必须周期性地转储数据库，这也加重了 DBA 的负担。为避免出现介质故障而影响数据库的可用性，许多 DBMS 提供了数据库镜像(Mirror)的功能用于数据库恢复。

(二)根据 DBA 的要求，DBMS 自动把整个数据库或关键数据复制到另一个磁盘上。当主数据库更新时，DBMS 自动把更新后的数据复制过去，即 DBMS 自动保证镜像数据与主数据的一致性。出现介质故障时，可由镜像磁盘继续提供使用，同时 DBMS 自动利用镜像磁盘数据进行数据库的恢复，不需要关闭系统和重装数据库副本；没有出现故障时，数据库镜像还可用于并发操作，即当一个用户对数据加排他锁修改数据时，其他用户可以读镜像数据库上的数据，而不必等待该用户释放锁。

本次授课小结

本次授课讲述了事务的概念和性质、数据库故障的三种类型、数据库恢复的技术和策略、数据库镜像的概念。

学生课后复习时应着重于其中的第 1、3、5 点内容，为进一步学习后续章节打好基础

实验

综合实验一