

# 数据库的恢复

浙江农林大学

主讲：刘丽娟

# 第十章 数据库恢复技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

10.1 事务的基本概念

10.2 数据库恢复概述

10.3 故障的种类

10.4 恢复的实现技术

10.5 恢复策略

10.6 具有检查点的恢复技术

10.7 数据库镜像

10.8 小结



## 10.2 数据库恢复概述

- 故障是不可避免的
  - 计算机硬件故障
  - 软件的错误
  - 操作员的失误
  - 恶意的破坏
- 故障的影响
  - 运行事务非正常中断，影响数据库中数据的正确性
  - 破坏数据库，全部或部分丢失数据



# 数据库恢复概述（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 数据库的恢复
  - 数据库管理系统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(亦称为一致状态或完整状态)的功能，这就是数据库的恢复管理系统对故障的对策
- 恢复子系统是数据库管理系统的一个重要组成部分
- 恢复技术是衡量系统优劣的重要指标



# 第十章 数据库恢复技术

- 10.1 事务的基本概念
- 10.2 数据库恢复概述
- 10.3 故障的种类
- 10.4 恢复的实现技术
- 10.5 恢复策略
- 10.6 具有检查点的恢复技术
- 10.7 数据库镜像
- 10.8 小结



# 故障的种类



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

1.事务内部的故障

2.系统故障

3.介质故障

4.计算机病毒



# 1.事务内部的故障

- 事务内部的故障
  - 有的是可以通过事务程序本身发现的(见下面转账事务的例子)
  - 有的是非预期的，不能由事务程序处理的。



# 事务内部的故障（续）

- 例如，银行转账事务，这个事务把一笔金额从一个账户甲转给另一个账户乙。

BEGIN TRANSACTION

    读账户甲的余额BALANCE;

    BALANCE=BALANCE-AMOUNT; /\*AMOUNT 为转账金额\*/

    IF(BALANCE < 0 ) THEN

        {打印‘金额不足，不能转账’;

            /\*事务内部可能造成事务被回滚的情况\*/

        ROLLBACK; /\*撤销刚才的修改，恢复事务\*/

    }

ELSE

    {读账户乙的余额BALANCE1;

        BALANCE1=BALANCE1+AMOUNT;

        写回BALANCE1;

    COMMIT;}





## 事务内部的故障（续）

- 这个例子所包括的两个更新操作要么全部完成要么全部不做。否则就会使数据库处于不一致状态，例如只把账户甲的余额减少了而没有把账户乙的余额增加。
- 在这段程序中若产生账户甲余额不足的情况，应用程序可以发现并让事务滚回，撤销已作的修改，恢复数据库到正确状态。



## 事务内部的故障（续）

- 事务内部更多的故障是非预期的，是不能由应用程序处理的。
    - 运算溢出
    - 并发事务发生死锁而被选中撤销该事务
    - 违反了某些完整性限制而被终止等
- 以后，事务故障仅指这类非预期的故障





# 事务内部的故障（续）

- 事务故障意味着
  - 事务没有达到预期的终点(COMMIT或者显式的ROLLBACK)
  - 数据库可能处于不正确状态。
- 事务故障的恢复：事务撤消（UNDO）
  - 强行回滚（ROLLBACK）该事务
  - 撤销该事务已经作出的任何对数据库的修改，使得该事务象根本没有启动一样



## 2.系统故障

- 系统故障

称为软故障，是指造成系统停止运转的任何事件，使得系统要重新启动。

- 整个系统的正常运行突然被破坏
- 所有正在运行的事务都非正常终止
- 不破坏数据库
- 内存中数据库缓冲区的信息全部丢失



# 系统故障的常见原因



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 特定类型的硬件错误（如CPU故障）
- 操作系统故障
- 数据库管理系统代码错误
- 系统断电



# 系统故障的恢复

- 发生系统故障时，一些尚未完成的事务的结果可能已送入物理数据库，造成数据库可能处于不正确状态。
  - 恢复策略：系统重新启动时，恢复程序让所有非正常终止的事务回滚，强行撤消（UNDO）所有未完成事务





# 系统故障的恢复

- 发生系统故障时，有些已完成的事务可能有一部分甚至全部留在缓冲区，尚未写回到磁盘上的物理数据库中，系统故障使得这些事务对数据库的修改部分或全部丢失
  - 恢复策略：系统重新启动时，恢复程序需要重做（REDO）所有已提交的事务



### 3.介质故障

- 介质故障

称为硬故障，指外存故障

- 磁盘损坏
  - 磁头碰撞
  - 瞬时强磁场干扰
- 介质故障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
  - 介质故障比前两类故障的可能性小得多，但破坏性大得多





## 4.计算机病毒

- 计算机病毒
  - 一种人为的故障或破坏，是一些恶作剧者研制的一种计算机程序
  - 可以繁殖和传播，造成对计算机系统包括数据库的危害
- 计算机病毒种类
  - 小的病毒只有20条指令，不到50B
  - 大的病毒像一个操作系统，由上万条指令组成



# 计算机病毒（续）

- 计算机病毒的危害
  - 有的病毒传播很快，一旦侵入系统就马上摧毁系统
  - 有的病毒有较长的潜伏期，计算机在感染后数天或数月才开始发病
  - 有的病毒感染系统所有的程序和数据
  - 有的只对某些特定的程序和数据感兴趣
- 计算机病毒已成为计算机系统的主要威胁，自然也是数据库系统的主要威胁
- 数据库一旦被破坏仍要用恢复技术把数据库加以恢复



# 故障小结



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 各类故障，对数据库的影响有两种可能性
  - 一是数据库本身被破坏
  - 二是数据库没有被破坏，但数据可能不正确，这是由于事务的运行被非正常终止造成的。

# 恢复



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 恢复操作的基本原理：冗余
  - 利用存储在系统别处的冗余数据来重建数据库中已被破坏或不正确的那部分数据
- 恢复的实现技术：复杂
  - 一个大型数据库产品，恢复子系统的代码要占全部代码的10%以上



# 第十章 数据库恢复技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 10.1 事务的基本概念
- 10.2 数据库恢复概述
- 10.3 故障的种类
- 10.4 恢复的实现技术**
- 10.5 恢复策略
- 10.6 具有检查点的恢复技术
- 10.7 数据库镜像
- 10.8 小结

## 10.4 恢复的实现技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

### 恢复机制涉及的关键问题

#### 1. 如何建立冗余数据

- 数据转储 (backup)
- 登记日志文件 (logging)

#### 2. 如何利用这些冗余数据实施数据库恢复



# 10.4 恢复的实现技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

## 10.4.1 数据转储

## 10.4.2 登记日志文件

## 10.4.1 数据转储



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

1.什么是数据转储

2.转储方法





# 1.什么是数据转储

- 转储是指数据库管理员定期地将整个数据库复制到磁带、磁盘或其他存储介质上保存起来的过程
- 备用的数据文本称为**后备副本(backup)**或**后援副本**



## 数据转储（续）

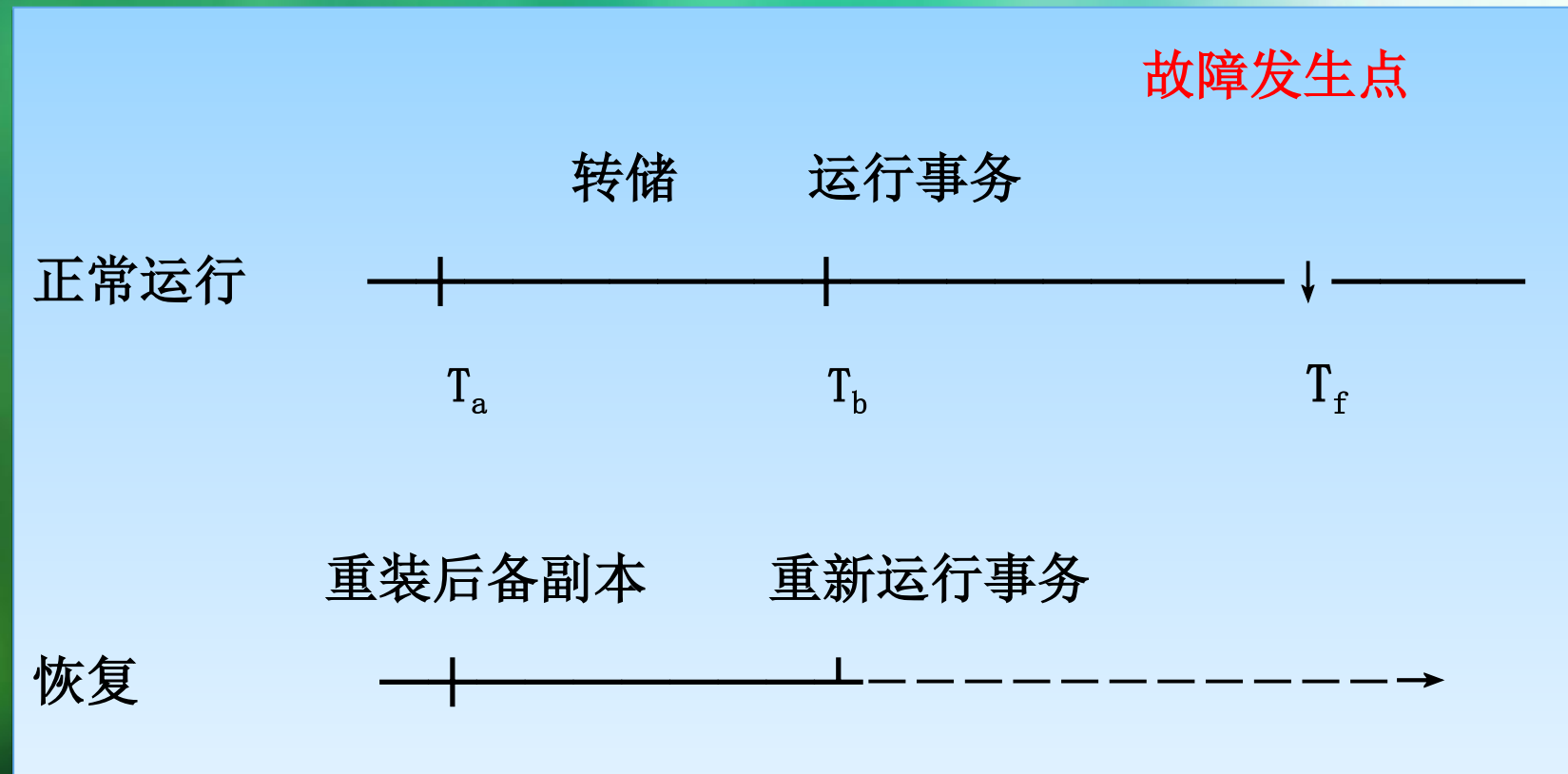
- 数据库遭到破坏后可以将后备副本重新装入
- 重装后备副本只能将数据库恢复到转储时的状态
- 要想恢复到故障发生时的状态，必须重新运行自转储以后的所有更新事务





# 数据转储（续）

[例]



转储和恢复



## 数据转储（续）

上图中：

- 系统在  $T_a$  时刻停止运行事务，进行数据库转储
- 在  $T_b$  时刻转储完毕，得到  $T_b$  时刻的数据库一致性副本
- 系统运行到  $T_f$  时刻发生故障
- 为恢复数据库，首先由数据库管理员重装数据库后备副本，将数据库恢复至  $T_b$  时刻的状态
- 重新运行自  $T_b \sim T_f$  时刻的所有更新事务，把数据库恢复到故障发生前的一致状态





## 2.转储方法

- (1) 静态转储与动态转储
- (2) 海量转储与增量转储
- (3) 转储方法小结



# （1）静态转储与动态转储

- 静态转储
  - 在系统中无运行事务时进行的转储操作
  - 转储开始时数据库处于一致性状态
  - 转储期间不允许对数据库的任何存取、修改活动
  - 得到的一定是一个数据一致性的副本
  - 优点：实现简单
  - 缺点：降低了数据库的可用性
    - 转储必须等待正运行的用户事务结束
    - 新的事务必须等转储结束





# 静态转储与动态转储（续）

- 动态转储
  - 转储操作与用户事务并发进行
  - 转储期间允许对数据库进行存取或修改
  - 优点
    - 不用等待正在运行的用户事务结束
    - 不会影响新事务的运行
  - 动态转储的缺点
    - 不能保证副本中的数据正确有效
    - 例在转储期间的某时刻  $T_c$ ，系统把数据  $A=100$  转储到磁带上，而在下一时刻  $T_d$ ，某一事务将  $A$  改为  $200$ 。  
后备副本上的  $A$  **过时**了

## 静态转储与动态转储（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 利用动态转储得到的副本进行故障恢复
  - 需要把动态转储期间各事务对数据库的修改活动登记下来，建立日志文件
  - 后备副本加上日志文件就能把数据库恢复到某一时刻的正确状态





## (2) 海量转储与增量转储

- 海量转储: 每次转储全部数据库
- 增量转储: 只转储上次转储后更新过的数据
- 海量转储与增量转储比较
  - 从恢复角度看, 使用海量转储得到的后备副本进行恢复往往更方便
  - 如果数据库很大, 事务处理又十分频繁, 则增量转储方式更实用更有效

### (3) 转储方法小结



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 转储方法分类

| 转储方式 | 转储状态   |        |
|------|--------|--------|
|      | 动态转储   | 静态转储   |
| 海量转储 | 动态海量转储 | 静态海量转储 |
| 增量转储 | 动态增量转储 | 静态增量转储 |



# 10.4 恢复的实现技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

## 10.4.1 数据转储

## 10.4.2 登记日志文件

## 10.4.2 登记日志文件



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

1.日志文件的格式和内容

2.日志文件的作用

3.登记日志文件





# 1.日志文件的格式和内容

- 什么是日志文件
  - 日志文件(log file)是用来记录事务对数据库的更新操作的文件
- 日志文件的格式
  - 以记录为单位的日志文件
  - 以数据块为单位的日志文件



## 日志文件的格式和内容（续）

- 以记录为单位的日志文件内容
  - 各个事务的开始标记(BEGIN TRANSACTION)
  - 各个事务的结束标记(COMMIT或ROLLBACK)
  - 各个事务的所有更新操作

以上均作为日志文件中的一个日志记录 (log record)





## 日志文件的格式和内容（续）

- 以记录为单位的日志文件，每条日志记录的内容
  - 事务标识（标明是哪个事务）
  - 操作类型（插入、删除或修改）
  - 操作对象（记录ID、Block NO.）
  - 更新前数据的旧值（对插入操作而言，此项为空值）
  - 更新后数据的新值（对删除操作而言，此项为空值）

# 日志文件的格式和内容（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 以数据块为单位的日志文件，每条日志记录的内容
  - 事务标识
  - 被更新的数据块





## 2.日志文件的作用

- 用途
  - 进行事务故障恢复
  - 进行系统故障恢复
  - 协助后备副本进行介质故障恢复



## 日志文件的作用（续）

- 具体作用
  - 事务故障恢复和系统故障恢复必须用日志文件。
  - 在动态转储方式中必须建立日志文件，后备副本和日志文件结合起来才能有效地恢复数据库。





## 日志文件的作用（续）

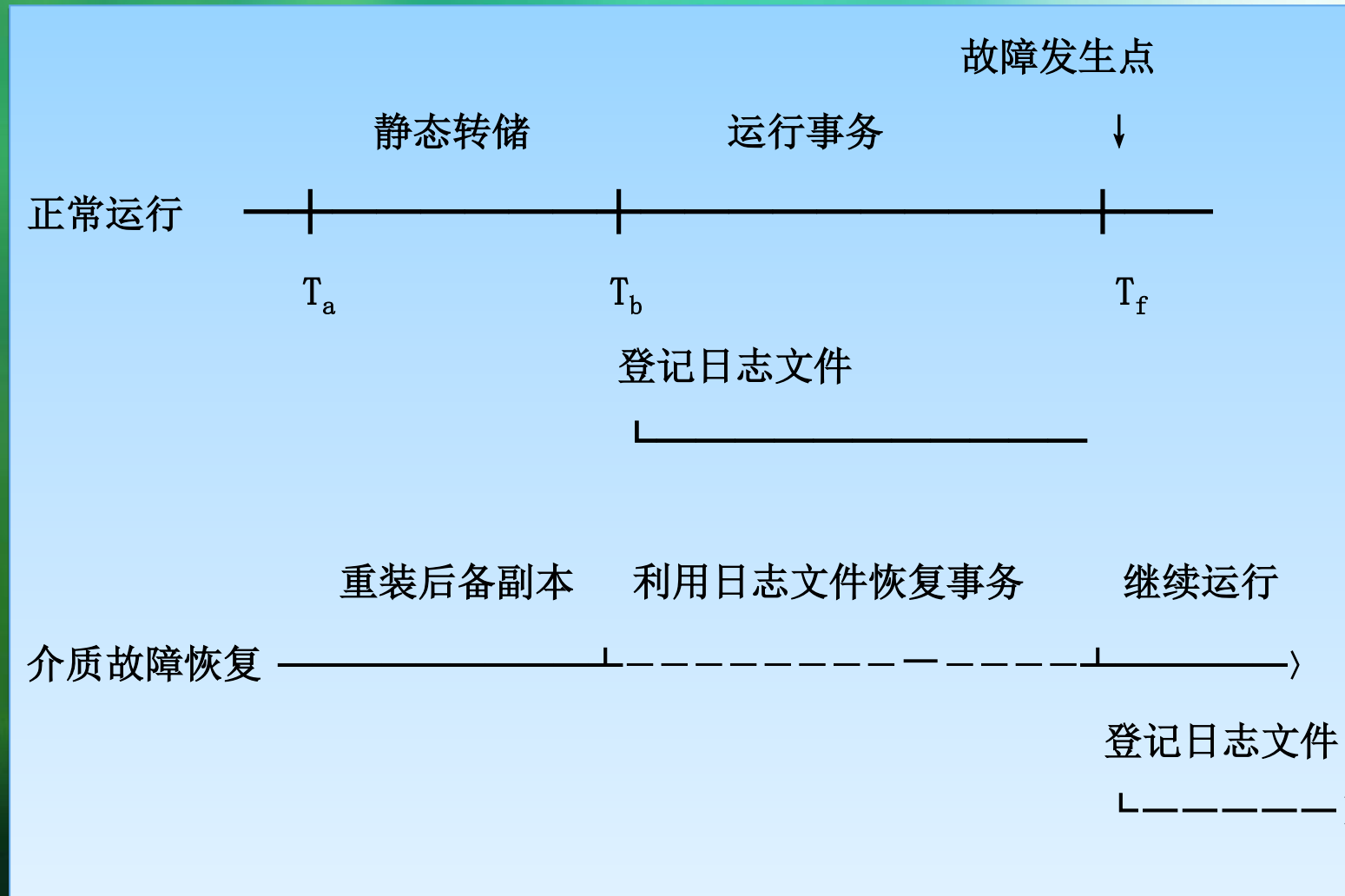
### — 在静态转储方式中，也可以建立日志文件。

- 当数据库毁坏后可重新装入后援副本把数据库恢复到转储结束时刻的正确状态
- 利用日志文件，把已完成的事务进行重做处理
- 对故障发生时尚未完成的事务进行撤销处理
- 不必重新运行那些已完成的事务程序就可把数据库恢复到故障前某一时刻的正确状态

# 日志文件的作用（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY



利用日志文件恢复





### 3.登记日志文件

- 为保证数据库是可恢复的，登记日志文件时必须遵循两条原则
  - 登记的次序严格按并发事务执行的时间次序
  - 必须先写日志文件，后写数据库
    - 写日志文件操作：把表示这个修改的日志记录写到日志文件中
    - 写数据库操作：把对数据的修改写到数据库中



# 登记日志文件（续）

- 为什么要先写日志文件
  - 写数据库和写日志文件是两个不同的操作
  - 在这两个操作之间可能发生故障
  - 如果先写了数据库修改，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
  - 如果先写日志，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



# 第十章 数据库恢复技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

10.1 事务的基本概念

10.2 数据库恢复概述

10.3 故障的种类

10.4 恢复的实现技术

10.5 恢复策略

10.6 具有检查点的恢复技术

10.7 数据库镜像

10.8 小结



## 10.5 恢复策略



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

### 10.5.1 事务故障的恢复

### 10.5.2 系统故障的恢复

### 10.5.3 介质故障的恢复





## 10.5.1 事务故障的恢复

- 事务故障：事务在运行至正常终止点前被终止
- 恢复方法
  - 由恢复子系统利用日志文件撤消（UNDO）此事务已对数据库进行的修改
- 事务故障的恢复由系统自动完成，对用户是透明的，不需要用户干预



# 事务故障的恢复步骤

- (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。即将日志记录中  
“更新前的值” 写入数据库。
  - 插入操作，“更新前的值” 为空，则相当于做删除操作
  - 删除操作，“更新后的值” 为空，则相当于做插入操作
  - 若是修改操作，则相当于用修改前值代替修改后值





## 事务故障的恢复步骤（续）

- （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。
- 参见爱课程网10.3节动画《事务故障及其恢复》

## 10.5 恢复策略



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

10.5.1 事务故障的恢复

10.5.2 系统故障的恢复

10.5.3 介质故障的恢复





## 10.5.2 系统故障的恢复

- 系统故障造成数据库不一致状态的原因
  - 未完成事务对数据库的更新可能已写入数据库
  - 已提交事务对数据库的更新可能还留在缓冲区  
没来得及写入数据库
- 恢复方法
  1. Undo 故障发生时未完成的事务
  2. Redo 已完成的事务
- 系统故障的恢复由系统在重新启动时自动完成，不需要用户干预



# 系统故障的恢复步骤

(1) 正向扫描日志文件（即从头扫描日志文件）

— 重做(REDO) 队列: 在故障发生前已经提交的事务

- 这些事务既有BEGIN TRANSACTION记录，也有COMMIT记录

— 撤销 (UNDO)队列:故障发生时尚未完成的事务

- 这些事务只有BEGIN TRANSACTION记录，无相应的COMMIT记录





# 系统故障的恢复步骤（续）

## （2）对撤销(UNDO)队列事务进行撤销(UNDO)处理

- 反向扫描日志文件，对每个撤销事务的更新操作执行逆操作
- 即将日志记录中“更新前的值”写入数据库

## （3）对重做(RED0)队列事务进行重做(RED0)处理

- 正向扫描日志文件，对每个重做事务重新执行登记的操作
- 即将日志记录中“更新后的值”写入数据库

- 参见爱课程网10.3节动画《系统故障及其恢复》

## 10.5 恢复策略



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

10.5.1 事务故障的恢复

10.5.2 系统故障的恢复

10.5.3 介质故障的恢复



## 10.5.3 介质故障的恢复



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

1.重装数据库

2.重做已完成的事务

## • 恢复步骤

- (1) 装入最新的后备数据库副本(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，利用恢复系统故障的方法（即REDO+UNDO），才能将数据库恢复到一致性状态。





## 介质故障的恢复（续）

(2) 装入有关的日志文件副本(转储结束时刻的日志文件副本)，重做已完成的事务。

- 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（续）

# 介质故障的恢复需要数据库管理员介入

- 数据库管理员的工作
  - 重装最近转储的数据库副本和有关的各日志文件副本
  - 执行系统提供的恢复命令
- 具体的恢复操作仍由数据库管理系统完成
- 参见爱课程网10.3节动画《介质故障及其恢复》



# 第十章 数据库恢复技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 10.1 事务的基本概念
- 10.2 数据库恢复概述
- 10.3 故障的种类
- 10.4 恢复的实现技术
- 10.5 恢复策略
- 10.6 具有检查点的恢复技术**
- 10.7 数据库镜像
- 10.8 小结



## 10.6 具有检查点的恢复技术



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

1.问题的提出

2.检查点技术

3.利用检查点的恢复策略





# 1.问题的提出

- 两个问题

- 搜索整个日志将耗费大量的时间
- 重做处理：重新执行，浪费了大量时间



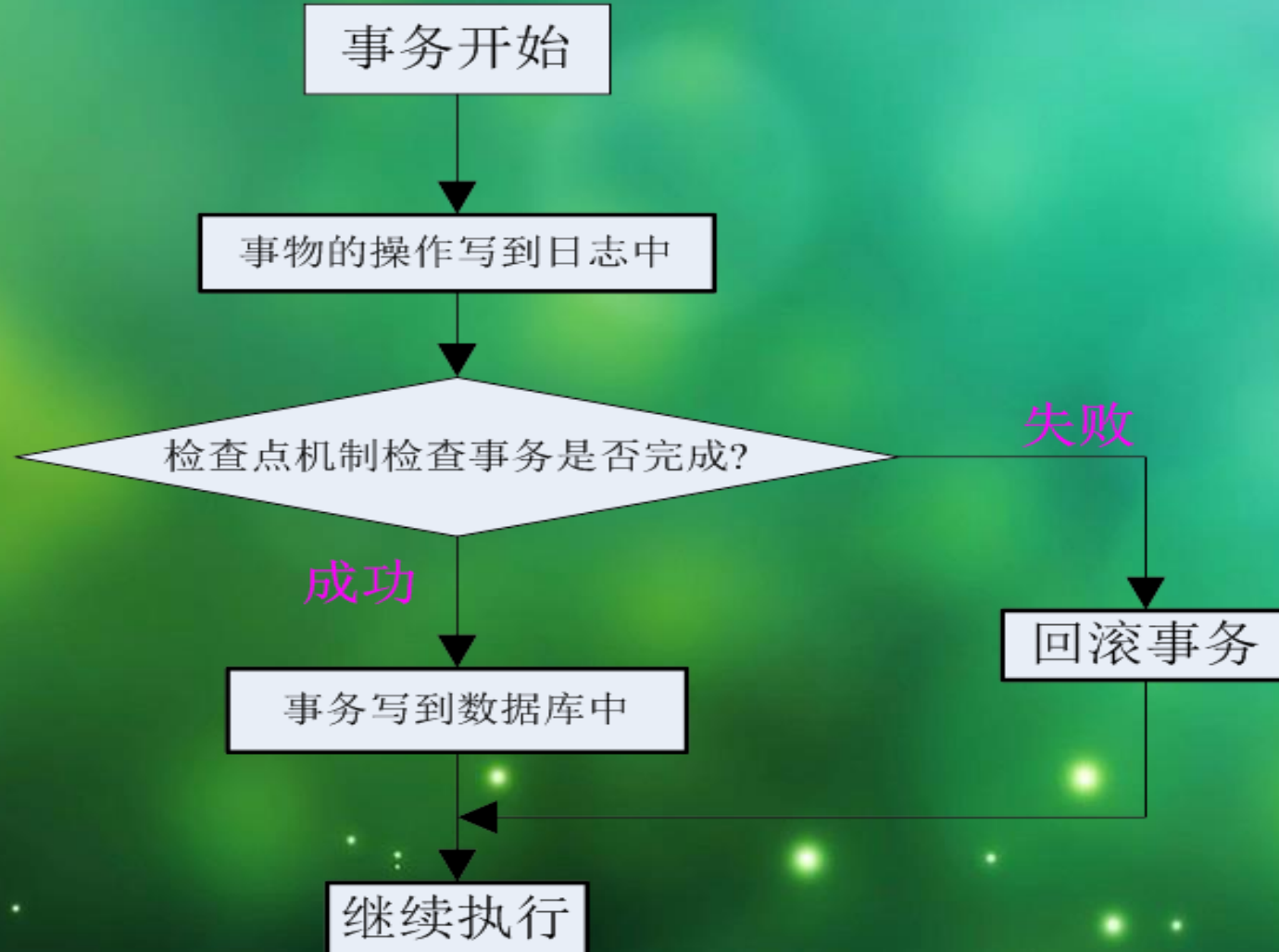
## 2.检查点技术

- 系统在检查点做的工作
  - 暂时终止现有事务的执行
  - 把主存中被修改的数据缓冲区强制写入磁盘
  - 在日志中写入检查点记录，并把日志强制写入磁盘
  - 重新开始执行事务





# 检查点技术（续）

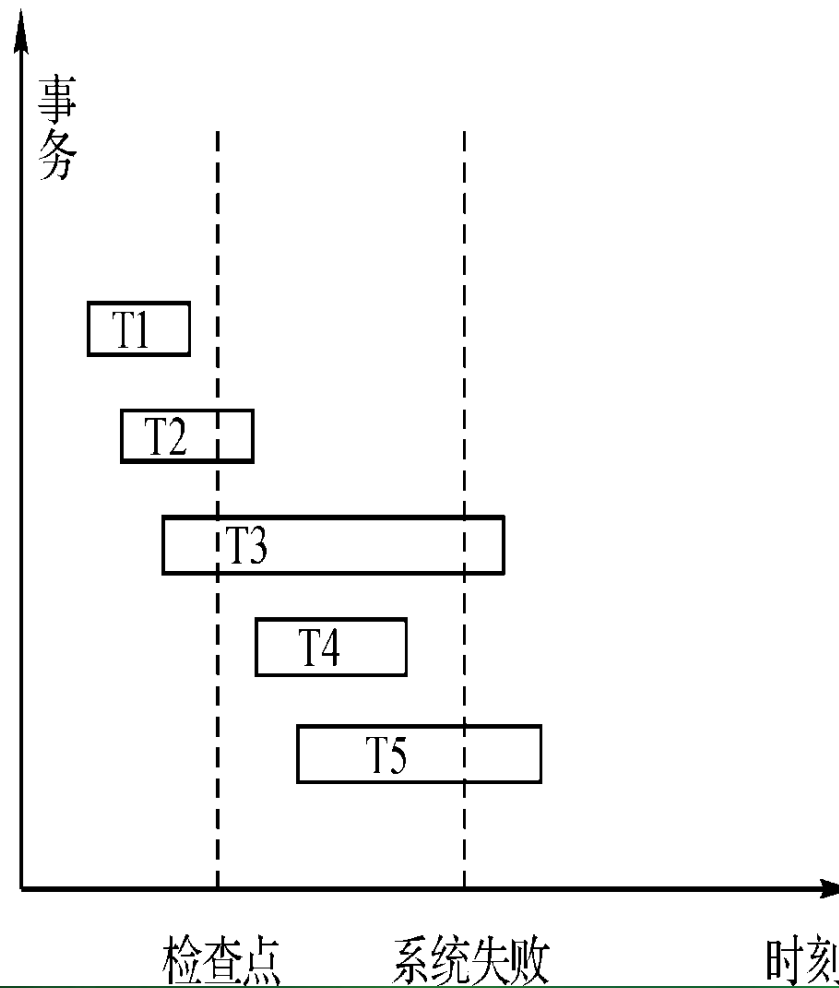


事务的工作原理

# 检查点技术（续）



浙江农林大学  
HEJIANG A&F UNIVERSITY



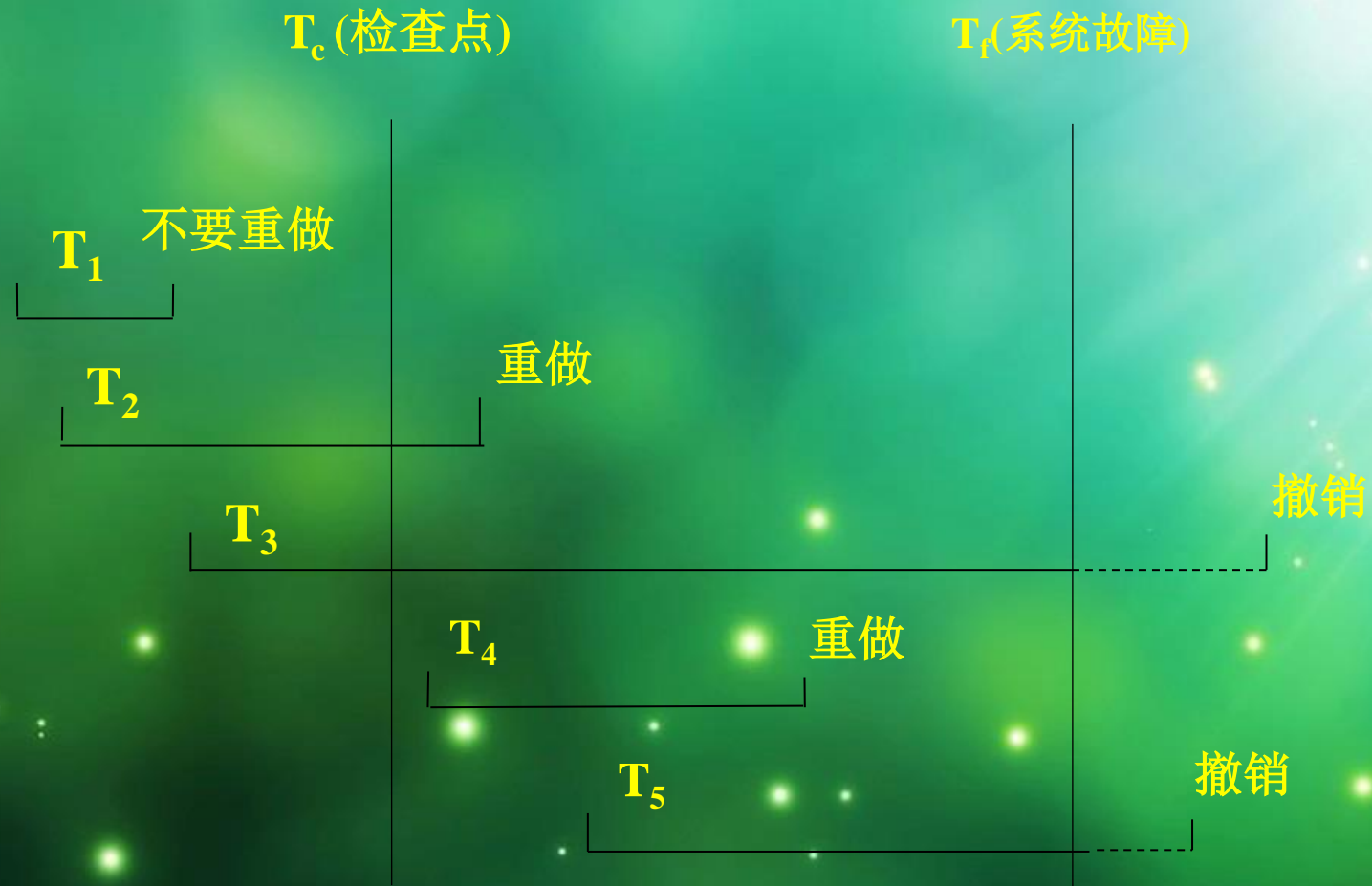
事务复原和检查点机制示意图





# 利用检查点的恢复策略（续）

系统出现故障时，恢复子系统将根据事务的不同状态采取不同的恢复策略



# 利用检查点的恢复策略（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- T1: 在检查点之前提交
- T2: 在检查点之前开始执行，在检查点之后故障点之前提交
- T3: 在检查点之前开始执行，在故障点时还未完成
- T4: 在检查点之后开始执行，在故障点之前提交
- T5: 在检查点之后开始执行，在故障点时还未完成





# 利用检查点的恢复策略（续）

## 恢复策略

- T3和T5在故障发生时还未完成，所以予以撤销
- T2和T4在检查点之后才提交，它们对数据库所做的修改在故障发生时可能还在缓冲区中，尚未写入数据库，所以要重做
- T1在检查点之前已提交，所以不必执行重做操作



# 第十章 数据库恢复技术

10.1 事务的基本概念

10.2 数据库恢复概述

10.3 故障的种类

10.4 恢复的实现技术

10.5 恢复策略

10.6 具有检查点的恢复技术

10.7 数据库镜像

10.8 小结





## 10.7 数据库镜像

- 介质故障是对系统影响最为严重的一种故障，严重影响数据库的可用性
  - 介质故障恢复比较费时
  - 为预防介质故障，数据库管理员必须周期性地转储数据库
- 提高数据库可用性的解决方案
  - 数据库镜像（Mirror）

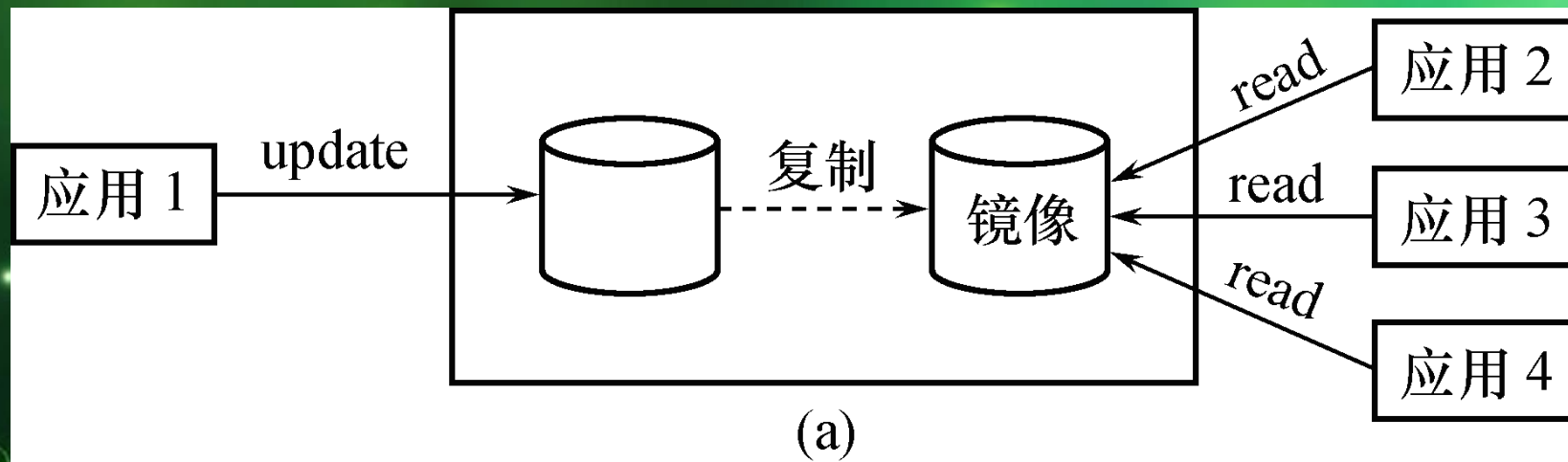


## 数据库镜像（续）

- 数据库镜像

- 数据库管理系统自动把整个数据库或其中的关键数据复制到另一个磁盘上
- 数据库管理系统自动保证镜像数据与主数据的一致性

每当主数据库更新时，数据库管理系统自动把更新后的数据复制过去

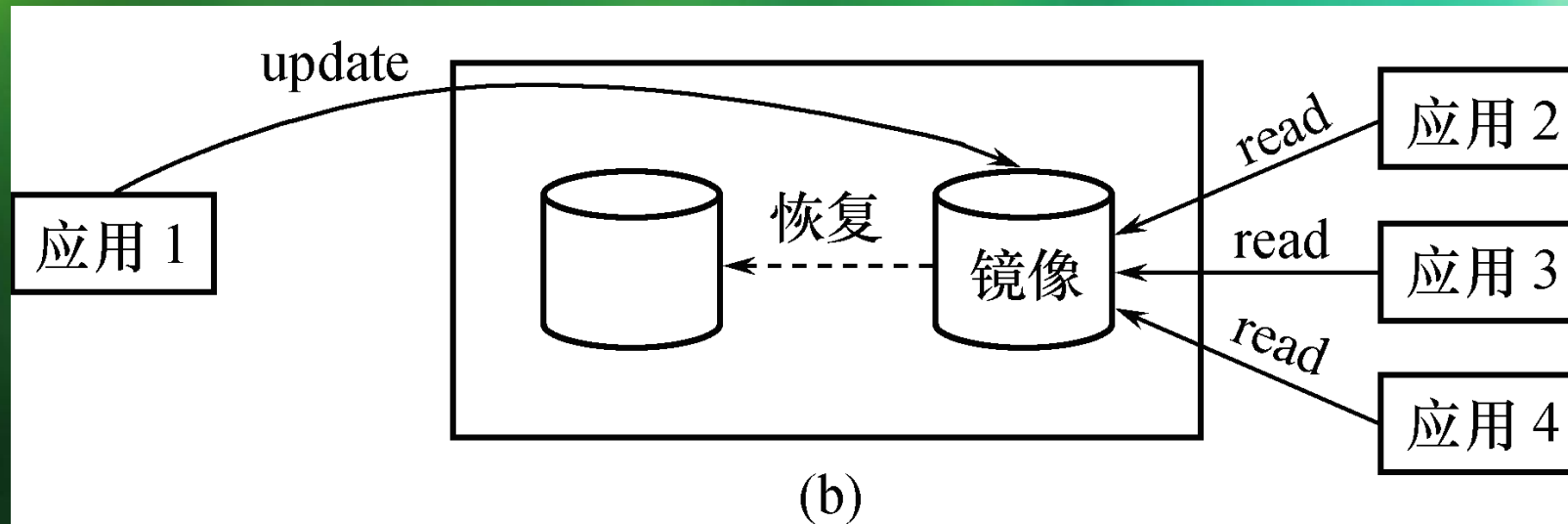






# 数据库镜像的用途

- 出现介质故障时
  - 可由镜像磁盘继续提供使用
  - 同时数据库管理系统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

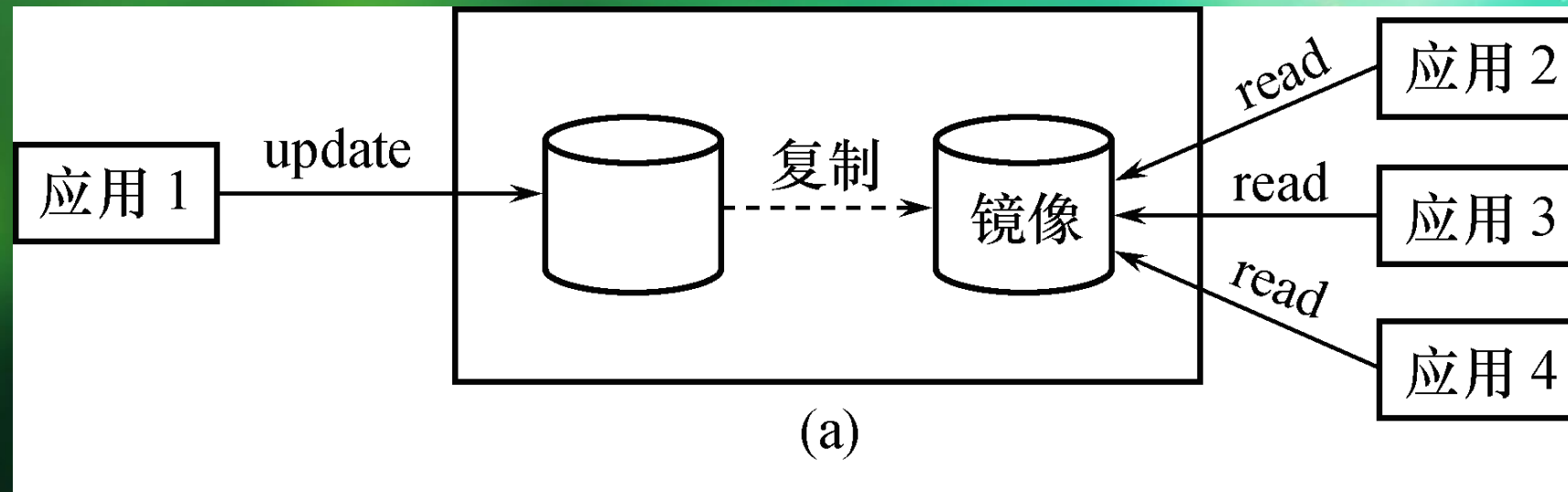




## 数据库镜像（续）

### ❖ 没有出现故障时

- 可用于并发操作
- 一个用户对数据加排他锁修改数据，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





# 数据库镜像（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 频繁地复制数据自然会降低系统运行效率
  - 在实际应用中用户往往只选择对**关键数据**和**日志文件**镜像
    - 不是对整个数据库进行镜像



# 创建备份和还原计划，要考虑下面的问题

- 备份什么类型的数据库？

系统和用户数据库经常有不同的备份和还原需要。

master:创建数据库、改变了配置值、配置数据库登录名等等，才需要备份。

处理实时客户事务的关键用户数据库：每小时备份一次。

- 数据库中数据的重要性如何？

对于每个工作日都要更新的数据库，每周执行2次完整备份，并且使用每天差异备份以及每小时备份事务日志作为补充。

- 数据库多久发生一次改变？

只读数据库不需要定期备份，连续不停更新的数据库应该使用连续备份。





# 创建备份和还原计划，要考虑下面的问题

- 需要还原数据库的速度有多快？

担负关键任务的数据库，需要迅速恢复到联机状态，需要使用磁盘驱动器或者多个备份设备。

- 可以压缩备份吗？

压缩和解压缩可能会显著增加处理器开销，会对服务器性能产生显著影响。

- 计划备份的最佳时间？

使用频率尽可能低时进行备份。





# SQL Server基本备份类型

- 完整备份

包括所有对象、系统表以及数据。复制数据库中的一切，而且还包括备份进行过程中的事务日志部分。

- 差异备份

用于备份自最近一次完整备份之后发生改变的数据。

因为只保存改变内容，所以这种类型的备份比较快，可以更频繁地执行。也包含了事务日志部分。

- 事务日志备份

备份将存储自上一次事务日志备份后发生的改变，不同于完整和差异备份，记录备份操作开始时的事务日志状态（而不是结束时的状态）



# SQL Server基本备份类型



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 文件和文件组备份

可以备份数据库文件和文件组而不是备份整个数据库。如果正在处理大型数据库，并且希望只备份各个文件而不是整个数据库以节省时间，那么这种备份是有用的。



# 三种恢复模式

- 简单

恢复到上一次备份点而设计的，这种模式的备份策略由完整和差异备份策略组成。当启用这种模式时，不能使用事务日志备份。

- 完整

需要恢复到失败点或指定时间点的数据库。使用这种模式的备份策略应该包括完整、差异以及事务日志备份或者仅包含完整和事务日志。

- 大容量日志

减少日志空间的使用，但仍然保持完整恢复模式的大多数灵活性。

数据库属性→选项页--→恢复模式



# 硬件和媒体



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 磁带驱动器：盒式磁带，速度慢、成本低。
- 数字音频磁带（DAT）驱动器：最常见的是数字线性磁带（DLT）：35GB或40GB非压缩数据。
- 自动加载磁带系统：使用磁带库扩展备份量，通过在系统中使用多个驱动器，可以每小时记录数百GB的数据。
- 光学光盘柜：使用磁性的光学磁盘来提供高容量解决方案，缺点是成本高。
- 可移动磁盘
- 磁盘驱动器



# 第十章 数据库恢复技术

- 10.1 事务的基本概念
- 10.2 数据库恢复概述
- 10.3 故障的种类
- 10.4 恢复的实现技术
- 10.5 恢复策略
- 10.6 具有检查点的恢复技术
- 10.7 数据库镜像
- 10.8 小结**



## 10.8 小结



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 事务的概念和性质
  - 事务是数据库的逻辑工作单位
  - 数据库管理系统保证系统中一切事务的原子性、一致性、隔离性和持续性，就保证了事务处于一致状态

# 小结（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 故障的种类
  - 事务故障
  - 系统故障
  - 介质故障
- 恢复中最经常使用的技术
  - 数据库转储
  - 登记日志文件



## 小结（续）



浙江农林大学  
ZHEJIANG A&F UNIVERSITY

- 恢复的基本原理
  - 利用存储在后备副本、日志文件和数据库镜像中的冗余数据来重建数据库
- 事务
  - 不仅是恢复的基本单位
  - 也是并发控制的基本单位