

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



10.6 具有检查点的恢复技术

- 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



解决方案

❖ 具有检查点（**checkpoint**）的恢复技术

- 在日志文件中增加检查点记录（**checkpoint**）
- 增加重新开始文件
- 恢复子系统在登录日志文件期间动态地维护日志



2.检查点技术

❖ 检查点记录的内容

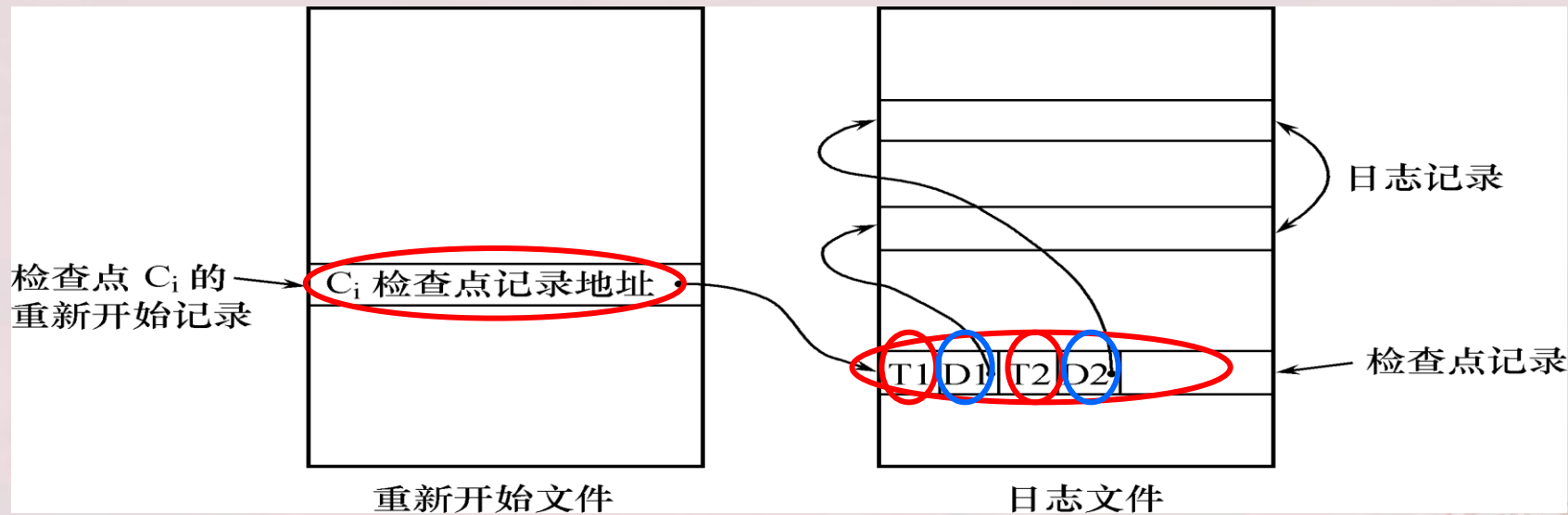
- 建立检查点时刻所有正在执行的事务清单
- 这些事务最近一个日志记录的地址

❖ 重新开始文件的内容

- 记录各个检查点记录在日志文件中的地址



检查点技术（续）



具有检查点的日志文件和重新开始文件



动态维护日志文件的方法

❖ 动态维护日志文件的方法

周期性地执行如下操作：建立检查点，保存数据库状态。

具体步骤是：

- (1) 将当前日志缓冲区中的所有日志记录写入磁盘的日志文件上
- (2) 在日志文件中写入一个检查点记录
- (3) 将当前数据缓冲区的所有数据记录写入磁盘的数据库中
- (4) 把检查点记录在日志文件中的地址写入一个重新开始文件



建立检查点

❖ 恢复子系统可以定期或不定期地建立检查点,保存数据库状态

■ 定期

- 按照预定的一个时间间隔, 如每隔一小时建立一个检查点

■ 不定期

- 按照某种规则, 如日志文件已写满一半建立一个检查点

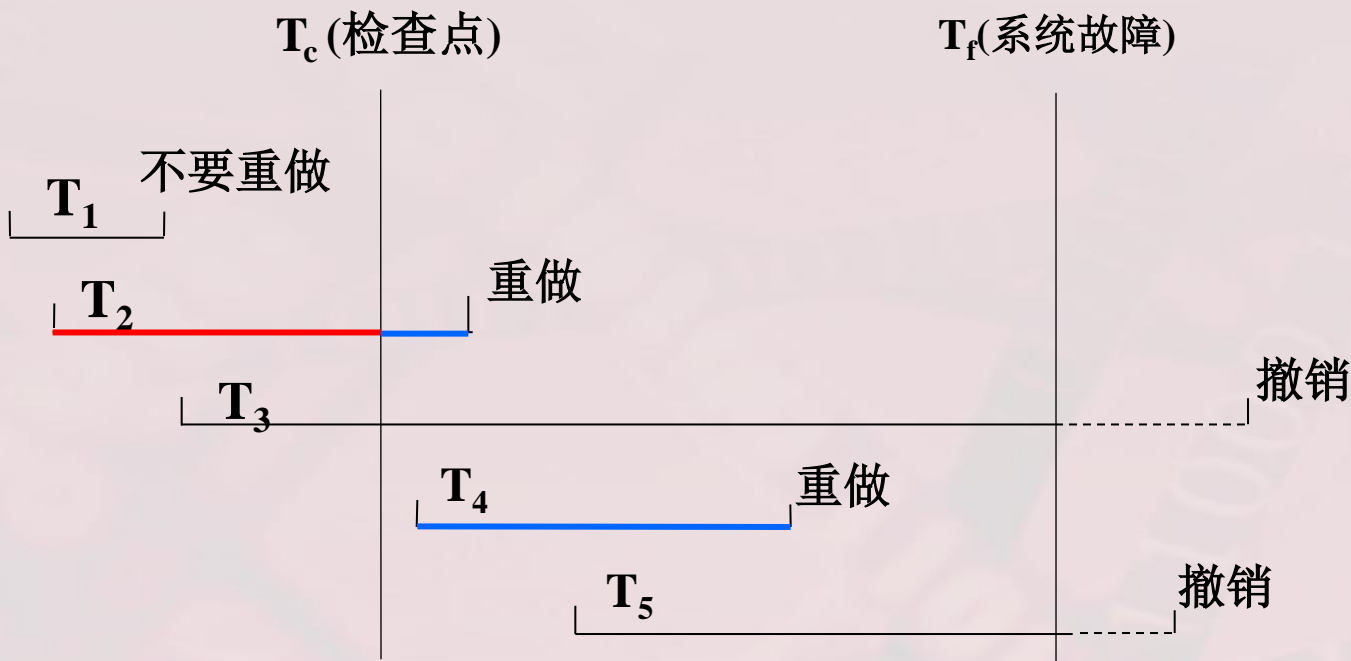


3.利用检查点的恢复策略

- ❖ 使用检查点方法可以改善恢复效率
 - 当事务T在一个检查点之前提交
 - T对数据库所做的修改已写入数据库
 - 写入时间是在这个检查点建立之前或在这个检查点建立之时
 - 在进行恢复处理时，没有必要对事务T执行重做操作
 - 当事务T在检查点时还没有完成
 - T对数据库所做的修改已写入数据库
 - 在进行恢复处理时，如果需要重做T，重做的起始点是检查点。



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
在日志文件中的地址，由该地址在日志文件中找
到最后一个检查点记录



利用检查点的恢复策略（续）

(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**

- 建立两个事务队列

 - **UNDO-LIST**

 - **REDO-LIST**

- 把**ACTIVE-LIST**暂时放入**UNDO-LIST**队列，**REDO**队列暂为空。



利用检查点的恢复策略（续）

- (3) 从检查点开始正向扫描日志文件，直到日志文件结束
- 如有新开始的事务 T_i ，把 T_i 暂时放入**UNDO-LIST**队列
 - 如有提交的事务 T_j ，把 T_j 从**UNDO-LIST**队列移到**REDO-LIST**队列;直到日志文件结束
- (4) 对**UNDO-LIST**中的每个事务执行**UNDO**操作
对**REDO-LIST**中的每个事务执行**REDO**操作



小结

❖ 具有检查点的恢复技术

- 为什么要引入检查点
- 如何建立检查点
- 利用检查点的恢复策略



思考题

❖ 我们学习了建立检查点的4个步骤

- 1) 将当前日志缓冲区中的所有日志记录写入磁盘的日志文件上
- 2) 在日志文件中写入一个检查点记录
- 3) 将当前数据缓冲区的所有数据记录写入磁盘的数据库中
- 4) 把检查点记录在日志文件中的地址写入一个重新开始文件

如果在建立检查点过程中发生了系统故障，恢复子系统会如何恢复数据库？



