西安交通大学
软件学院

操作系统原理

Operating System Principle

田丽华

§7-4 死锁避免

Deadlock Avoidance
（死锁避免）

01 允许进程动态地申请资源，系统在进行资源分配之前，先计算资源分配的安全性

02 若此次分配不会导致系统从安全状态向不安全状态转换，便可将资源分配给进程；否则不分配资源，进程必须阻塞等待。

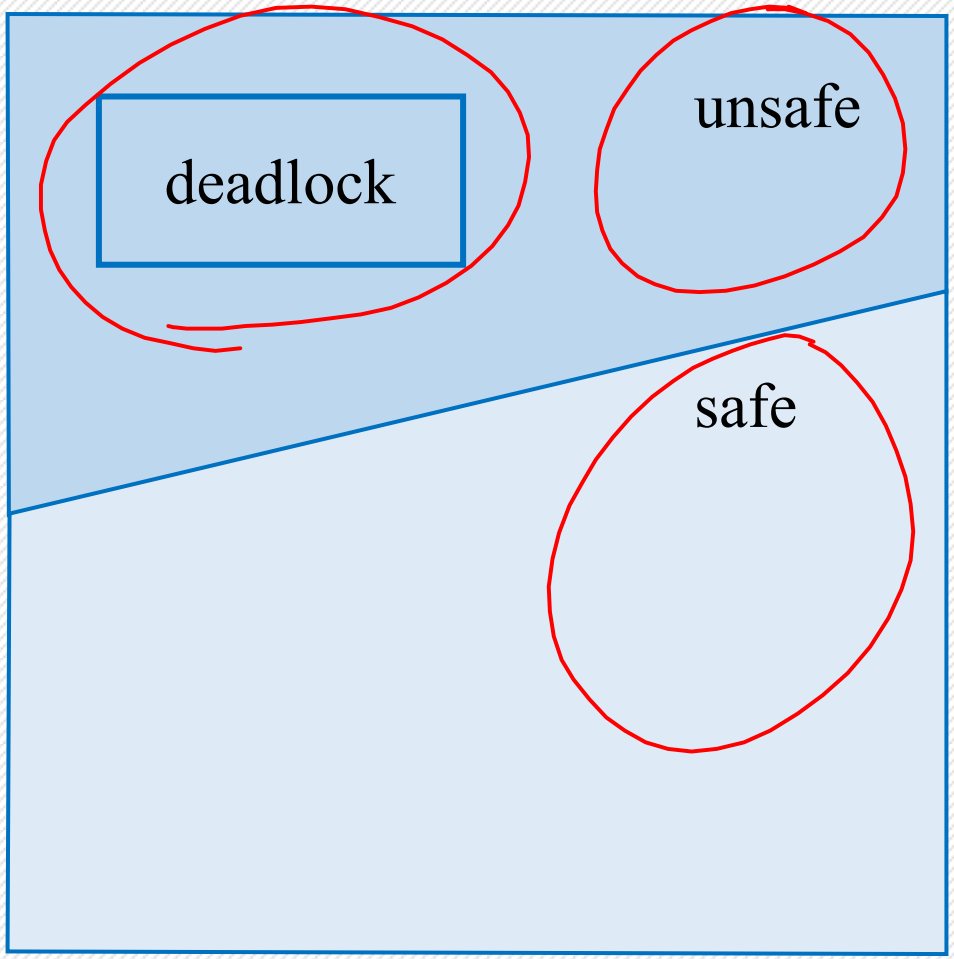
Methods for Handling Deadlocks
处理死锁的方法

Safe State

- **安全状态**是指系统的一种状态，在此状态下,系统能按某种顺序（例如P_1、P_2……P_n）来为各个进程分配其所需资源，直至最大需求，使每个进程都可顺序地一个个地完成。这个序列（P_1、P_2……P_n）称为**安全序列**。
- System is in safe state if there exists a safe sequence of all processes. （如果存在一个安全序列系统处于安全态）
- 若某一时刻不存在一个安全序列，则称系统处于不安全状态。

Safe, unsafe , deadlock state spaces

安全、不安全、死锁状态空间



Basic Facts
基本事实

- If a system is in safe state ⇒ no deadlocks.

  (如果一个系统在安全状态，就没有死锁)

  if a system is deadlock ⇒ in unsafe state

  (如果系统死锁，则处于不安全状态)

- If a system is in unsafe state ⇒ possibility of deadlock.

  (如果一个系统处于不安全状态，就有可能死锁)

- Avoidance ⇒ ensure that a system will never enter an unsafe state.

  (避免：确保系统永远不会进入不安全状态)

Basic Facts
基本事实

- Single instance of a resource type
  资源有单个实例
  - Use a resource-allocation graph资源分配图
- Multiple instances of a resource type
  资源有多个实例
  - Use the banker's algorithm银行家算法

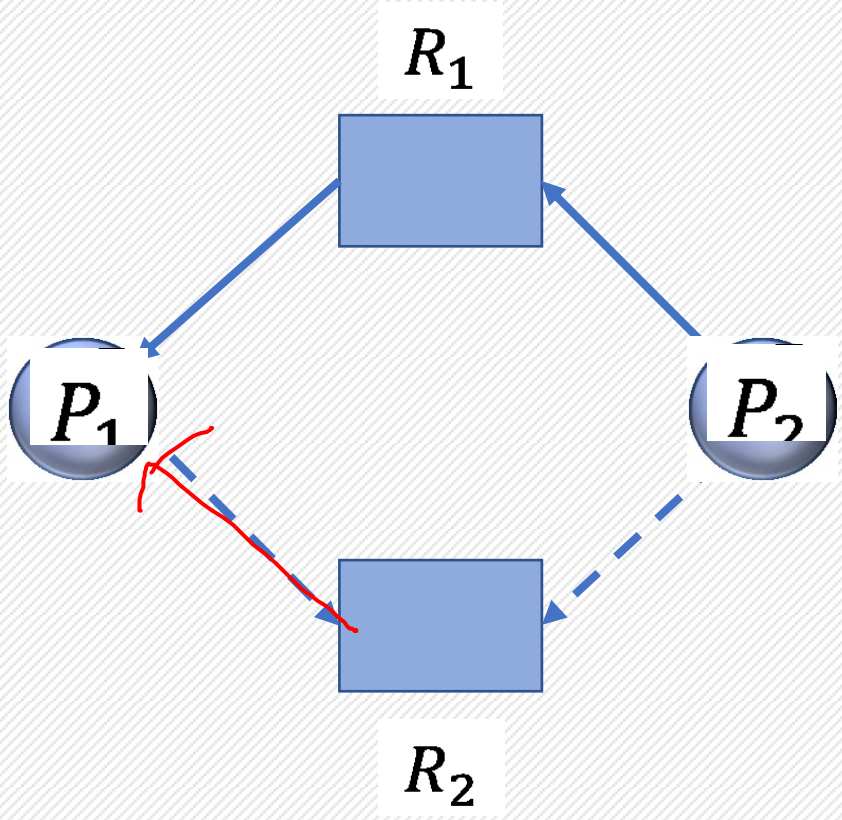
Resource-Allocation Graph Algorithm
资源分配图算法

- **Claim edge $P_i \rightarrow R_j$ indicated that process P_j may request resource R_j; represented by a dashed line.(claim edge 需求边$P_i \rightarrow R_j$代表进程Pi可能会申请资源Ri，表示为虚线)**
- **Claim edge converts to request edge when a process requests a resource.（一个进程申请资源的时候，需求边转化为请求边）**
- **When a resource is released by a process, assignment edge reconverts to a claim edge.（当资源被进程释放的时候，分配边转化为需求边）**
- **Resources must be claimed a priori in the system.（系统中的资源必须被事先声明）**

Resource-Allocation Graph Algorithm
资源分配图算法

当一个进程Pi 申请资源Rj时，由循环检测算法来检查：

如果把图中的需求边$Pi \rightarrow Rj$转为分配边$Rj \rightarrow Pi$，图中是否会出现环路，只有不出现环路，才实施资源分配。



Unsafe State In A Resource-Allocation Graph

不安全的状态图