

# **SISTEMI OPERATIVI**

Sistemi Distribuiti  
Computazione Distribuita

## **Lezione 5 – Coordinamento distribuito tra processi (parte 1)**

**Vincenzo Piuri**

---

Università degli Studi di Milano

# Sommario

- Ordinamento degli eventi
- Sincronizzazione
  - Mutua esclusione
  - Atomicità
  - Gestione concorrenza

# **Sincronizzazione dei processi**

## **Ordinamento degli eventi in sistema distribuito**

- Soluzione ideale:  
ordinamento totale
- Problema:  
in sistemi distribuiti non ci sono  
orologio e memoria comune  
per realizzare ordinamento totale
- Soluzione pratica:  
ordinamento parziale

# Relazione “accaduto prima” (1)

## Definizione

- Se A e B sono eventi dello stesso processo ed A avviene prima di B,  
allora  $A \rightarrow B$
- Se A è l'evento di trasmissione del messaggio in un processo,  
e B è l'evento di ricezione di quel messaggio da parte di un altro processo,  
allora  $A \rightarrow B$
- Se  $A \rightarrow B$  e  $B \rightarrow C$ ,  
allora  $A \rightarrow C$

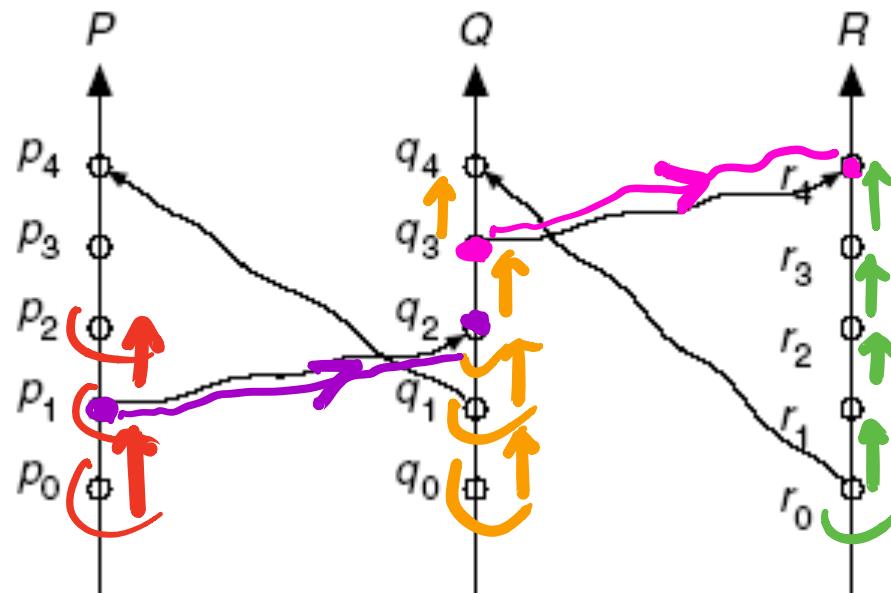
# **Relazione “accaduto prima” (2)**

## **Caratteristiche**

- Non riflessiva
- Eventi non in relazione sono concorrenti e non si influenzano
- Se due eventi sono in relazione possono influenzarsi

# Relazione “accaduto prima” (3)

## Tre processi concorrenti



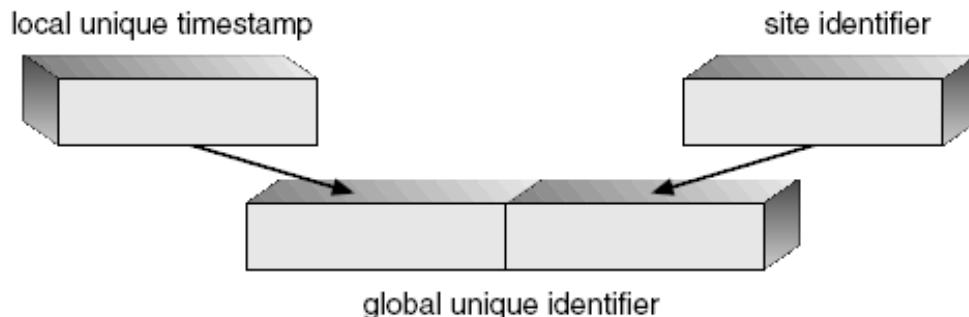
# **Ordinamento globale**

- Marca di tempo
- Orologio logico
- Incremento monotono
- Avanzamento forzato per processi diversi che comunicano

# Marca di tempo

## Generazione delle marche di tempo uniche a livello di sistema distribuito

- Soluzione centralizzata: unico distributore
- Soluzione distribuita:  
marca di tempo locale + identificatore del sito



# **Marca di tempo locale**

- Orologio di sistema
- Orologio logico locale

# Orologi locali

- Velocità diverse
- Orologi logici
- Sincronizzazione degli orologi logici
  - Aggiornamento dell'orologio a  $n+1$  qualora un sito riceva una marca di valore  $n$

# **Mutua esclusione (1)**

- Metodo centralizzato
- Metodo distribuito

# Mutua esclusione (2)

## Metodo centralizzato

- Processo coordinatore centralizzato per la gestione dell'accesso alle sezioni critiche
- Coda dei processi in attesa
- Prestazioni: limitate per centralizzazione
- Tolleranza ai guasti: vulnerabile

# Mutua esclusione (3)

## Metodo distribuito

- Quando P vuole entrare in sezione critica:
  - genera marca tempo
  - invia richiesta di entrata in sezione critica a tutti i processi
- Quando un processo Q riceve richiesta di entrare in sezione critica:
  - ritarda la risposta se è in sezione critica
  - risponde immediatamente se non intende entrare in sezione critica
  - se desidera entrare nella propria sezione critica ma non vi è entrato, compara la propria marca di tempo con quella di P: se la propria è più grande, allora risponde immediatamente, altrimenti ritarda la risposta per entrare prima

# Mutua esclusione (4)

## Metodo distribuito

- No starvation
- No deadlock
- Tolleranza ai guasti

# **Mutua esclusione (5)**

## **Metodo a passaggio di token**

- Processi organizzati ad anello logico
- Token indica autorizzazione ad accedere