CENTRO UNIVERSITÁRIO UNIMETROCAMP | WYDEN

Sistemas Distribuídos – Sincronização

Santiago A. Robles - santiago.robles@unimetrocamp.edu.br

março/2019

Por quê sincronização ?

- Colaboração na comunicação entre entidades de um SD

  Noção de Tempo em um SD:

  - Desvios de relógios
  - Falhas
  - atrasos

Sincronismo X Assincronismo

- Síncronos
  - Variação do tempo está delimitada
  - Sincronizar relógios
  - Limita o uso de concorrência
- Assíncronos
  - Variação do tempo não é limitada
  - Nada pode se assumir sobre os intervalos de tempo
  - Concorrência. Complicado detectar falhas de comunicação.

- HÍBRIDOS !

- Coordenação = a coisa certa
  - Coordenar ações e dados

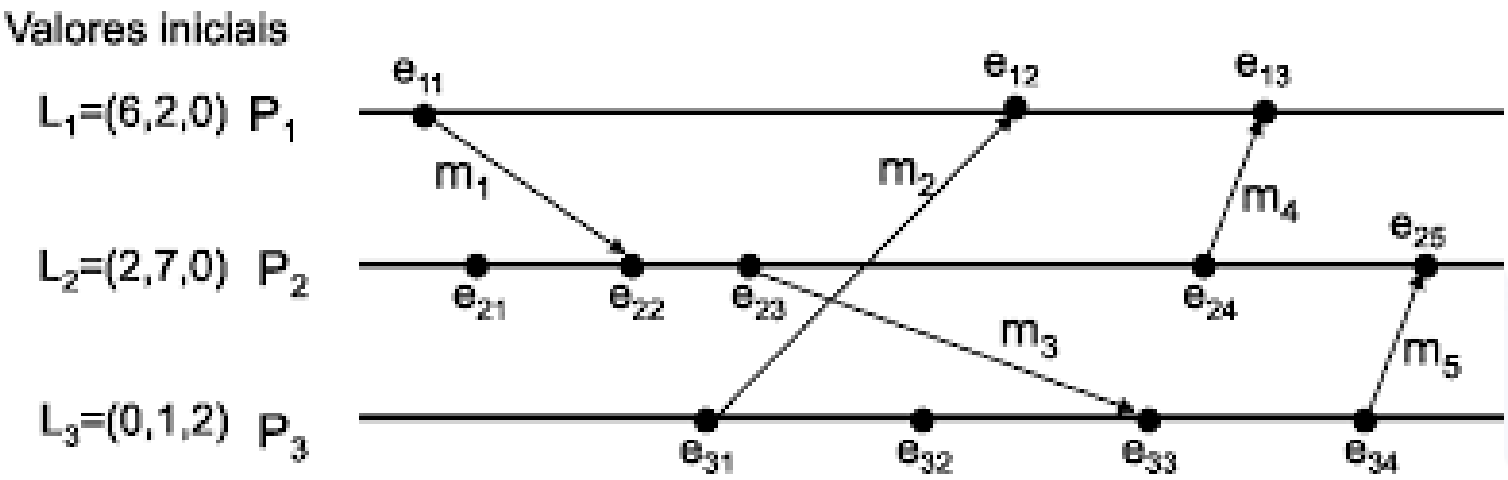
- Sincronização = no momento certo
  - Requer noção de tempo
  - Ordenar ações
  - Ordenar acesso aos recursos

Relógio Lógico X Relógio Físico

- Lógico
  - Sincronizar relógios de diferentes máquinas (relógio central)
  - NTP (Network Time Protocol)
- Físico
  - Ordem de execução
  - Relação “acontece-antes”

Relação “acontece-antes”

Ordem de execução !



e11 → e22
e22 → e23
e23 → e33
e11 → e33
e23 || e12

Algortimo Lamport

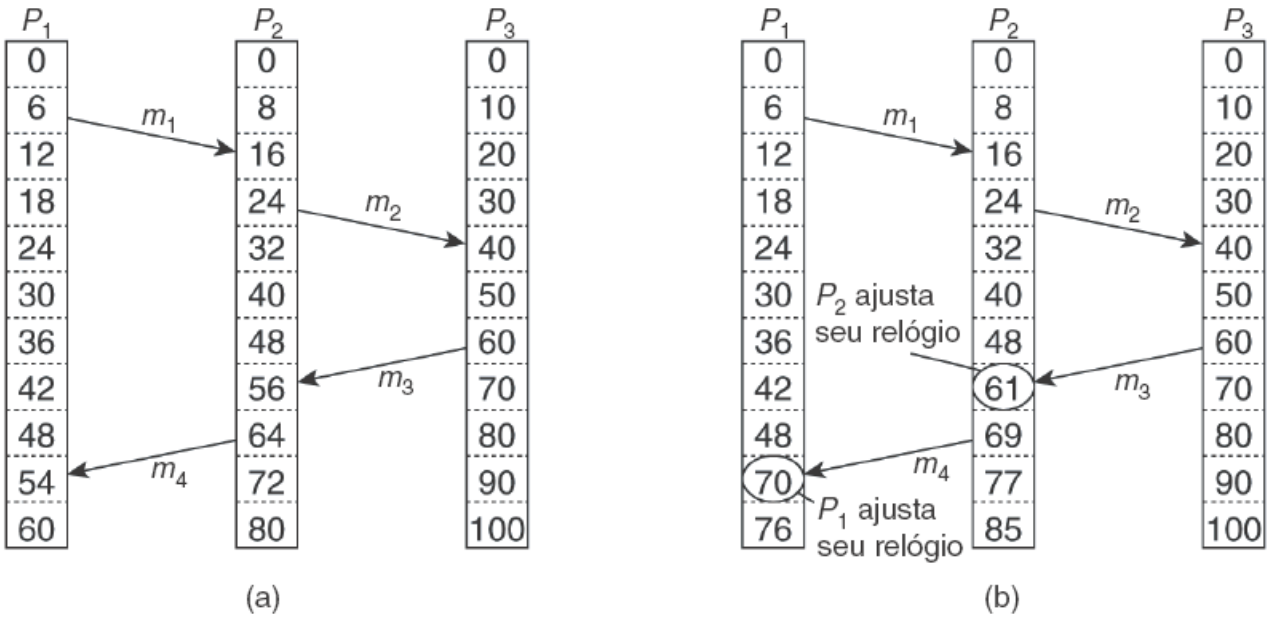


Figura 6.9 (a) Três processos, cada um com seu próprio relógio. Os relógios funcionam a taxas diferentes. (b) O algoritmo de Lamport corrige os relógios.

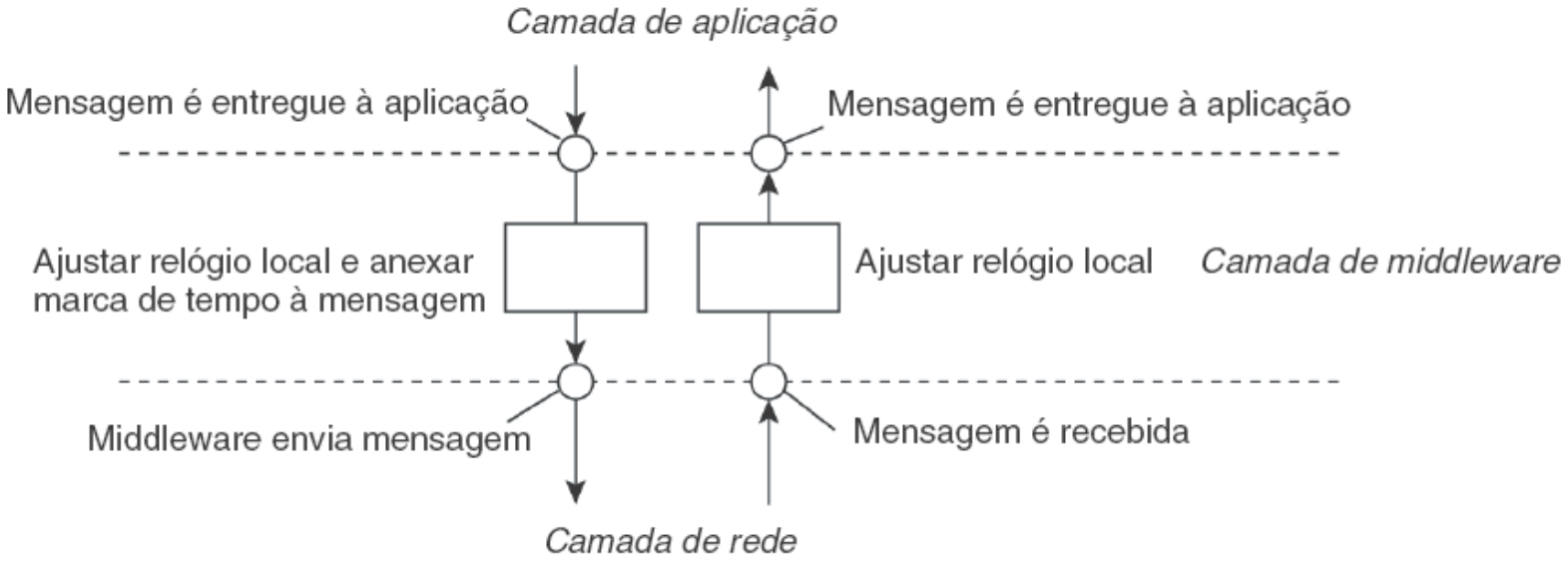


Figura 6.10 Posicionamento de relógios lógicos de Lamport em sistemas distribuídos.

Concorrência em um Sistema NÂO-Distribuído

- Evitar “race codintion”
- Assegurar Exclusão Mútua
  - Proteger regiões críticas
  - Locks ; Semáforos ; Monitores
- Evitar Deadlocks e Starvation

Mais desafios

- Não compartilham memória
- Não compartilham relógio
- Concorrência

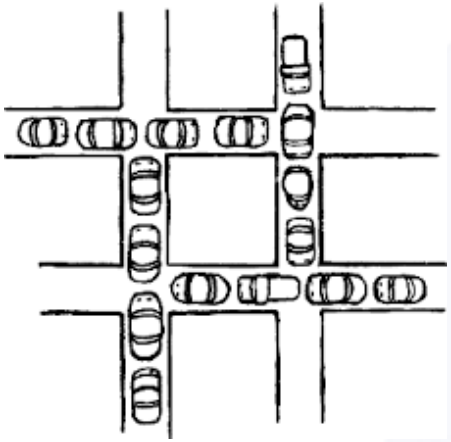
Exclusão Mútua

- Como evitar **Condições de Corrida (Race Condition)** ?
  - **Exclusão Mútua:** Impedir que 2 ou mais processos acessem um mesmo recurso simultâneamente
  - Garantia de acesso exclusivo a um recurso
  - A parte do programa em que um recurso compartilhado é acessado chama-se: **Região Crítica**
    - Uso de **Protocolos de Entrada e Saída**
  - Se um Processo está executando dentro da sua seção crítica, então nenhum outro processo pode estar executando em sua seção crítica correspondente.
    - Dois processos não podem estar simultaneamente dentro de suas regiões críticas correpondentes
  - Nenhum processo que esteja rodando fora de sua seção crítica pode bloquear a execução de outro processo

Starvation

- **Espera Limitada** - Nenhum processo pode ser obrigado a esperar indefinidamente para entrar em sua região crítica
  - Nenhuma suposição pode ser feita com relação à velocidade de execução dos processos ou ao número de processadores disponíveis no sistema

Deadlock



Deadlock é a situação em que um processo aguarda por um recurso que nunca estará disponível ou um evento que não ocorrerá. Para que ocorra a situação de deadlock, quatro condições são necessárias simultaneamente:

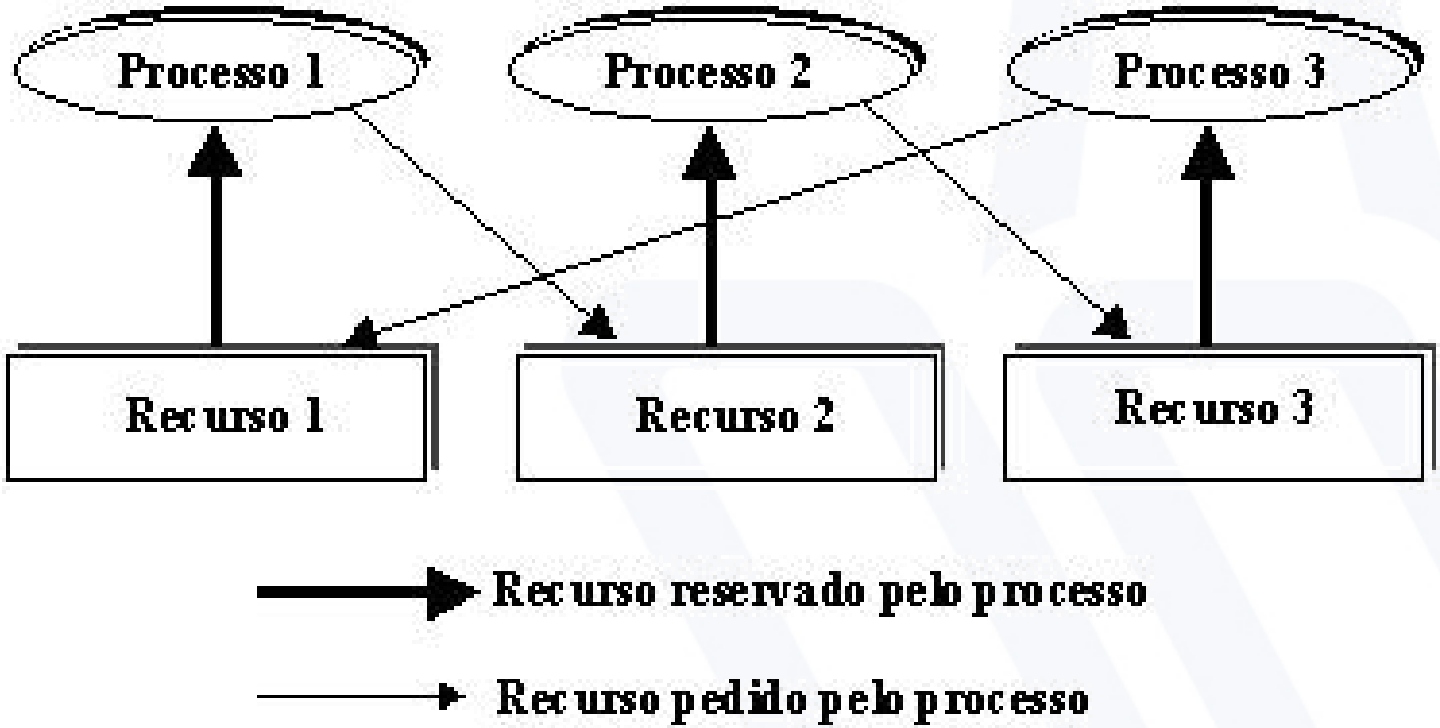
- exclusão mútua: cada recurso só pode estar alocado a um único processo em um determinado instante;

- espera por recurso: um processo, além dos recursos já alocados, pode estar esperando por outros recursos;

- não-preempção: um recurso não pode ser liberado de um processo só porque outros processos desejam o mesmo recurso;

- espera circular: um processo pode ter de esperar por um recurso alocado a outro processo e vice-versa.

UNI METROCAMP | WYDEN



Faci facid FACIMP FBV fmc Faculdade Martha Falcão ISL UNIFAVIP UNI METROCAMP RUY BARBOSA | AREA1 UniFBV UniFanor