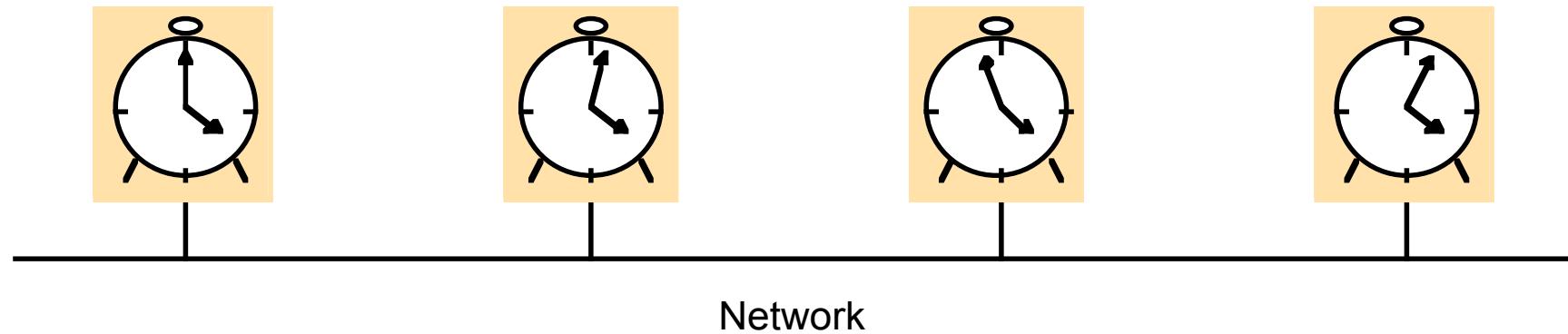


Processos, Relógios e Eventos

- Em computador com um processador, ordem dos eventos é bem definida:
 - $e_1 \rightarrow e_2$ se e_1 ocorre antes de e_2
 - Histórico: sequência de eventos $h = \langle e_1, e_2, \dots \rangle$ dentro de um mesmo processo
- Como atribuir um tempo a cada evento do sistema?

Relógio de Hardware Local

- Distorção - taxa de derivação



Sincronização de relógios físicos

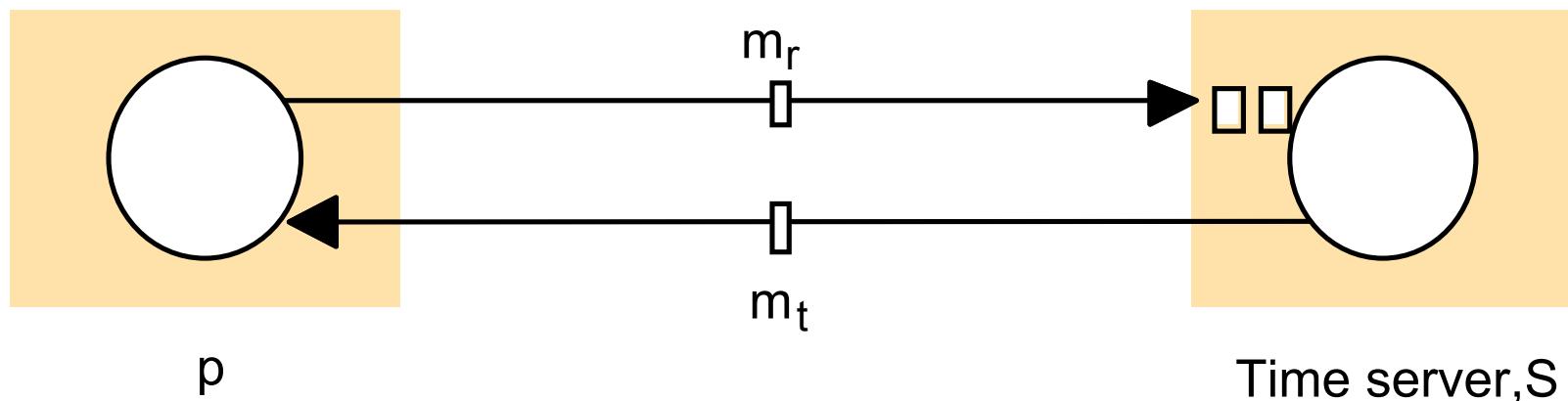
- UTC + GPS
- Sincronização Externa:
 - Dada fonte S e $D > 0$, $|S(t) - C_i(t)| < D$ para todo processo i
 - Relógios são precisos com limite D
- Sincronização Interna:
 - Dado $D > 0$, $|C_i(t) - C_j(t)| < D$ para todo par de processos (i,j)
 - Relógios concordam dentro do limite D

Sincronização de relógios físicos

- Correção
 - Taxa de derivação $< p$
 - Erro de medida do intervalo $t' - t$ é limitado
 - Monotonicidade
 - Se $t' > t$, então $C(t') > C(t)$
 - Dado valor de $H(t)$, podemos garantir monotonicidade popr software adequando os valores de α e β na equaçāo:
 - $C(t) = \alpha H(t) + \beta$

Sincronização de relógios físicos

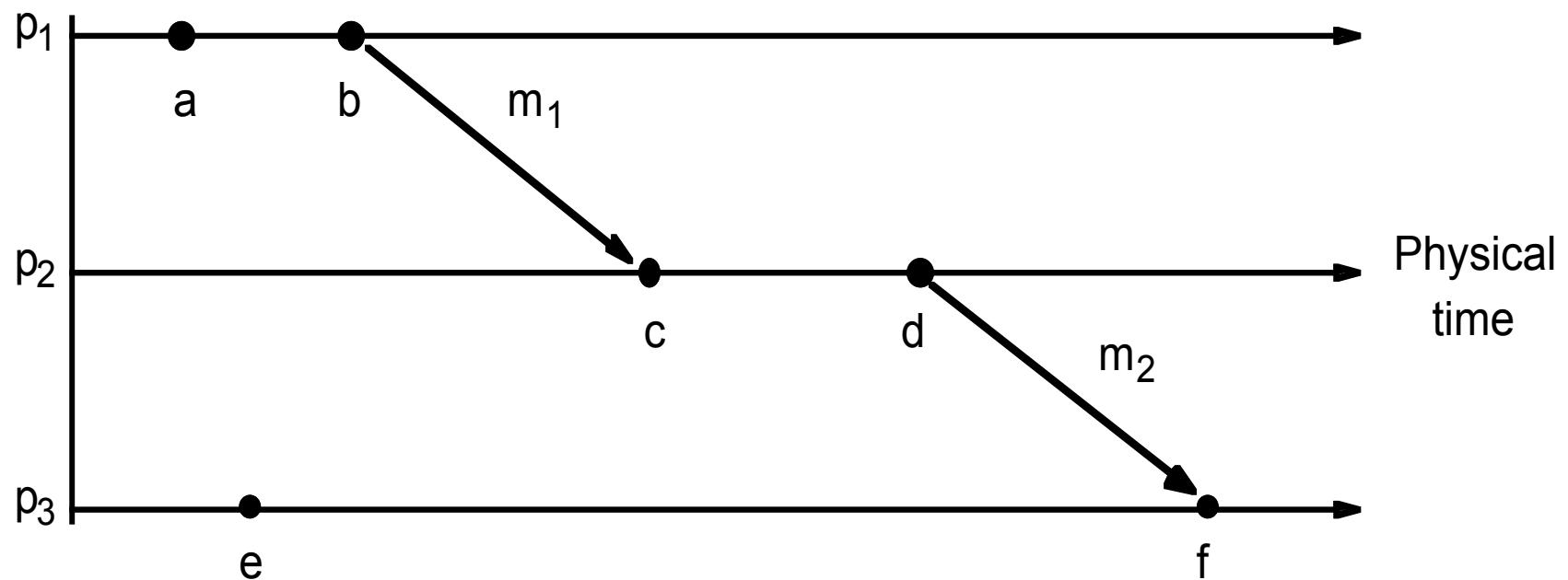
- Algoritmo de Berkeley
 - Mestre requisita tempos locais
 - Mestre faz média excluindo valores com diferença maior do que D
 - Mestre envia valor de ajuste (+/-) para cada processo



Tempo lógico, relógios lógicos

- Causalidade:
 - Quando dois eventos ocorrem no mesmo processo, a ordem é a mesma observada.
 - Quando uma mensagem é enviada, o evento *envio* ocorre antes do evento *recebe*.

Tempo lógico, relógios lógicos



Lamport78

- Relação Acontece Antes, \rightarrow (ordenação causal)
 - Se $e_1 \rightarrow_i e_2$ no processo i , então $e_1 \rightarrow e_2$
 - Para qualquer mensagem m , $\text{send}(m) \rightarrow \text{receive}(m)$
 - Se e_1, e_2 e e_3 obedecem $e_1 \rightarrow e_2$ e $e_2 \rightarrow e_3$ então $e_1 \rightarrow e_3$
 - Se $!(e_1 \rightarrow e_2)$ e $!(e_2 \rightarrow e_1)$, eventos são concorrentes

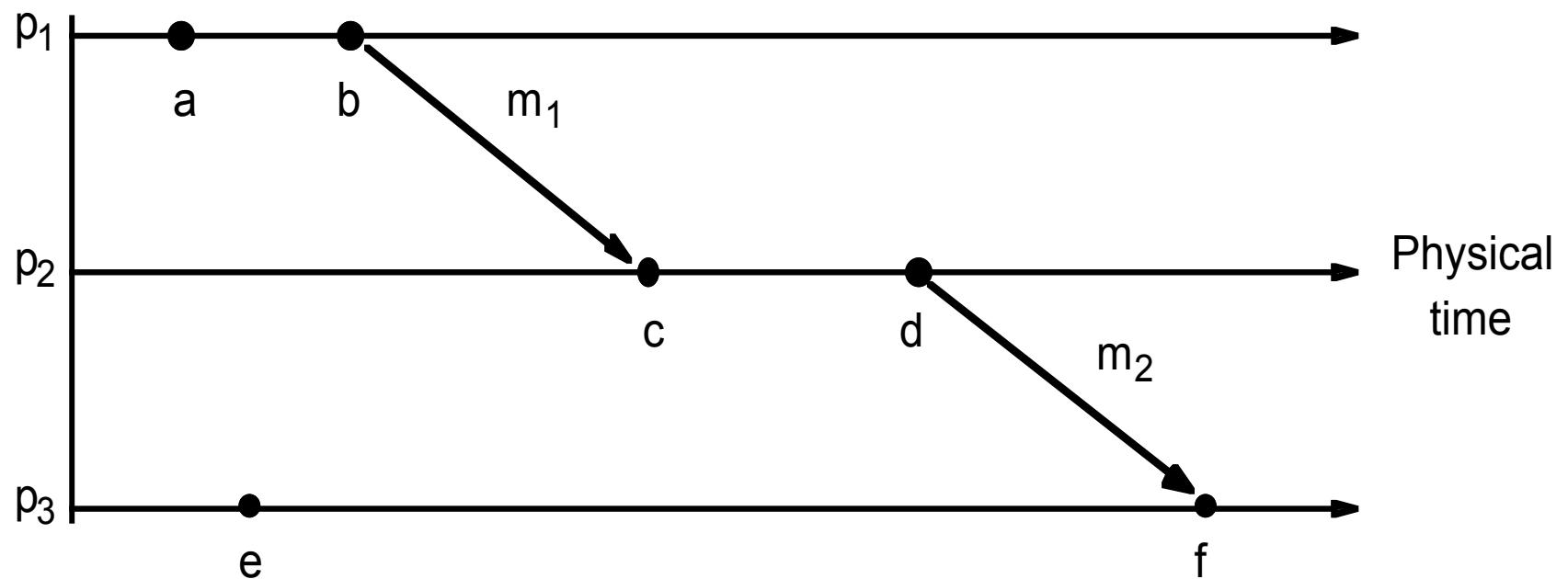
Como manter tempo lógico?

- Relógios lógicos
 - Cada processo mantém contador local L_i
 - Regra 1:
 - L_i é incrementado a cada evento no processo i .
 - Regra 2:
 - Quando processo envia mensagem m , inclui seu relógio lógico corrente
 - Na recepção de uma mensagem (m, t) , processo j calcula $L_j = 1 + \max(L_j, t)$ para determinar tempo de $\text{receive}(m)$

Como ordenar os eventos globalmente?

- Em caso de empate, impõe uma ordem arbitrária
- Evento X (em i) precede evento Y (em j) se e somente se
 - $L_i(X) < L_j(Y)$ ou
 - $L_i(X) == L_j(Y) \&\& P_i < P_j$
- Como usar??

Relógios Lógicos



Relógios Lógicos

