

# Sistemas Distribuídos e Tolerância a Falhas

## Tempo@Relógios

Baseado no trabalho de:  
Ana Sofia Cunha m1892  
Christian Lopes a15132

UBI, 2008

# O papel do tempo

2

- Crucial na ordenação de eventos
- Tempo Real?
  - Função monótona contínua e crescente
  - Unidade: segundo

*“Actualmente um segundo é a duração de 9.192.631.770 períodos da radiação emitida na transição entre dois níveis hiperfinos do estado fundamental do átomo de césio 133.”*

Graficamente pode ser representado por uma sequência de pontos sobre uma linha recta → ***Timeline***

# O papel do tempo

3

- O uso do tempo em sistemas distribuídos é feito em dois aspectos:
  - Registrar e observar a localização de eventos na timeline
    - ✦ Queremos saber qual a sequência em que ocorreu um conjunto de eventos (possivelmente distribuídos por várias máquinas)
  - Forçar o futuro posicionamento de eventos na timeline
    - ✦ Sincronização do progresso concorrente do sistema

# O papel do tempo

4

Para conhecermos qual a sequência de um conjunto de acontecimentos podemos marcar o instante de ocorrência atribuindo um,

- **Timestamp:** sequência de caracteres que marcam a data e/ou tempo no qual um certo evento ocorreu.

(ex. data de criação/alteração de um ficheiro)

- um timestamp está associado a um ponto na timeline.

Se queremos comparar a duração de vários acontecimentos podemos usar,

- **Intervalos de tempo:** cadeia de tempo composta por vários intervalos adicionados

# Durações Distribuídas

5

- **Timers e relógios locais:** implementam a abstracção da *timeline*.

Num sistema distribuído cada evento pode ocorrer em diferentes locais cada um com a sua *timeline*.

- Como conciliar diferentes *timelines*?
- Como medir durações distribuídas?

# Tempo Global vs Tempo Absoluto

6

- **Tempo Global (*global time*):** implementa a abstracção de um tempo universal, através de um relógio que fornece o mesmo tempo a todos os participantes no sistema.
- **Tempo Absoluto (*absolute time*):** padrões universalmente ajustados, disponíveis como fontes de tempo externo para o qual qualquer relógio interno se pode sincronizar.

# Relógios locais

7

- **Relógio físico local (physical clock - pc):** o modo mais comum para fornecer uma fonte de tempo num processo.
- O relógio num processo correcto,  $k$ , implementa uma função discreta, monótona crescente,  $pc_k$ , que mapeia o tempo real  $t$  em tempo de relógio  $pc_k(t)$ .
- Imperfeições de relógios físicos:
  - Granularidade ( $g$ )
  - Taxa de desvio ( $r$ )

# Propriedades de um relógio físico

8

- **Granularidade:** relógios físicos são granulares, isto é, avançam uma unidade em cada *tick*  $t_{tk}$ .

$$pc_k^{tk+1} - pc_k^{tk} = g$$

- **Taxa de desvio do relógio físico:** existe uma constante positiva  $r_p$ , a *taxa de desvio (rate of drift)*, que depende não só da qualidade do relógio mas também das condições ambientais (ex. temperatura).

$$0 \leq 1 - r_p \leq (pc_k(t_{tk+1}) - pc_k(t_{tk}))/g \leq 1 + r_p$$

$$\text{para } 0 \leq t_{tk} \leq t_{tk+1}$$

(Para uma taxa de desvio de  $10^{-5}$ , após 60 minutos o erro acumulado pode ser superior a 30 milisegundos)



# Para que serve um relógio local?

9

- Fornecer *timestamps* para eventos locais.
- Medir durações locais
  - O erro causado pelo desvio é normalmente insignificante para pequenas durações.
- Pode ser usado como um “timer” para estabelecer *timeouts*
- Medir durações distribuídas *round-trip*

# Relógios Globais - Características

10

- Fornecer o mesmo tempo para todos os intervenientes do sistema
- *Timestamping* de eventos distribuídos
- Medição de durações distribuídas

# Relógios Globais - funcionamento

11

1. É criado um relógio virtual  $vc_p$  para cada processo  $p$  a partir do relógio físico
2. É feita a sincronização de todos os relógios locais com o mesmo valor inicial  $vc_p(t_{init})$
3. Periodicamente os relógios virtuais são re-sincronizados



Algoritmo de Sincronização

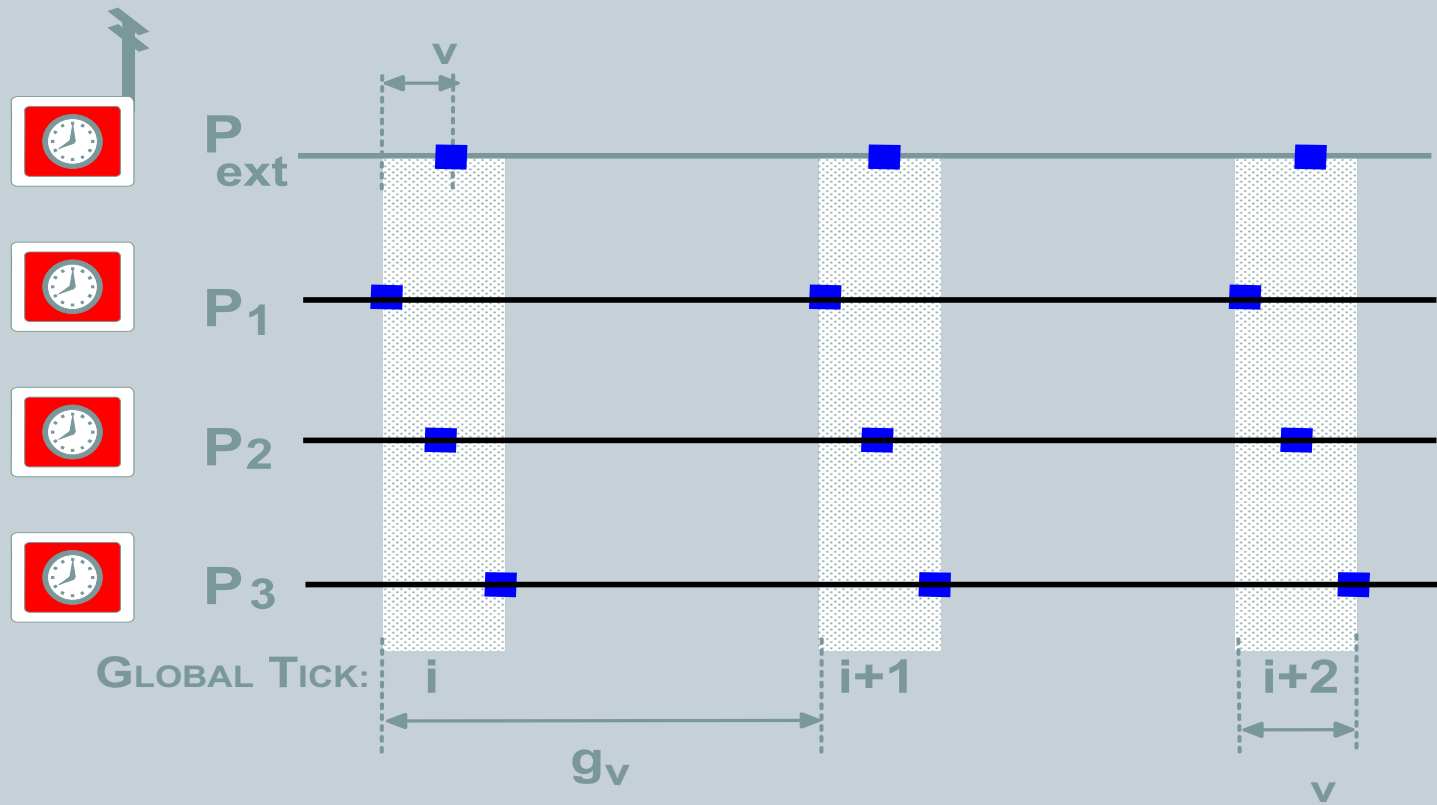
# Propriedades dos Relógios Globais

12

- Granularidade  $g_v = vc_p(t_{k+1}) - vc_p(t_k)$
- Precisão ( $\pi_v$ ): quão próximos os relógios se mantêm sincronizados entre si em qualquer instante do tempo.
- Exactidão ( $\alpha_v$ ): quão próximos os relógios estão sincronizados em relação a uma referência de tempo real absoluto (sincronização externa)

# Propriedades dos Relógios Globais (cont.)

13



# Sincronização interna vs Sincronização externa

14

- **Sincronização interna:**  
relógios tem que obter precisão relativamente a um tempo interno ao sistema
- **Sincronização externa:**  
relógios tem que estar sincronizados com uma fonte externa de tempo universal

# Referências de Tempo universal - Normas

15

- **Universal Time Coordinated (UTC)**

referência de tempo política, que inclui todas as propriedades usadas actualmente

- **Temps Atomic International (TAI)**

função contínua monotonicamente crescente a uma taxa constante (gerada por relógios atómicos de césio)

→ **Forma mais simples de obter o TAI:** por GPS (Global Position System) – é assegurada uma exactidão, em terra,  $\leq 100\text{ns}$  para os relógios dos receptores de GPS

# Medição de durações *round-trip*

16

- Certas durações distribuídas podem ser medidas sem a existência explícita de relógios globais
- O atraso de entrega de uma mensagem pode ser calculado com um erro conhecido e limitado, se existir uma mensagem prévia recente no sentido inverso



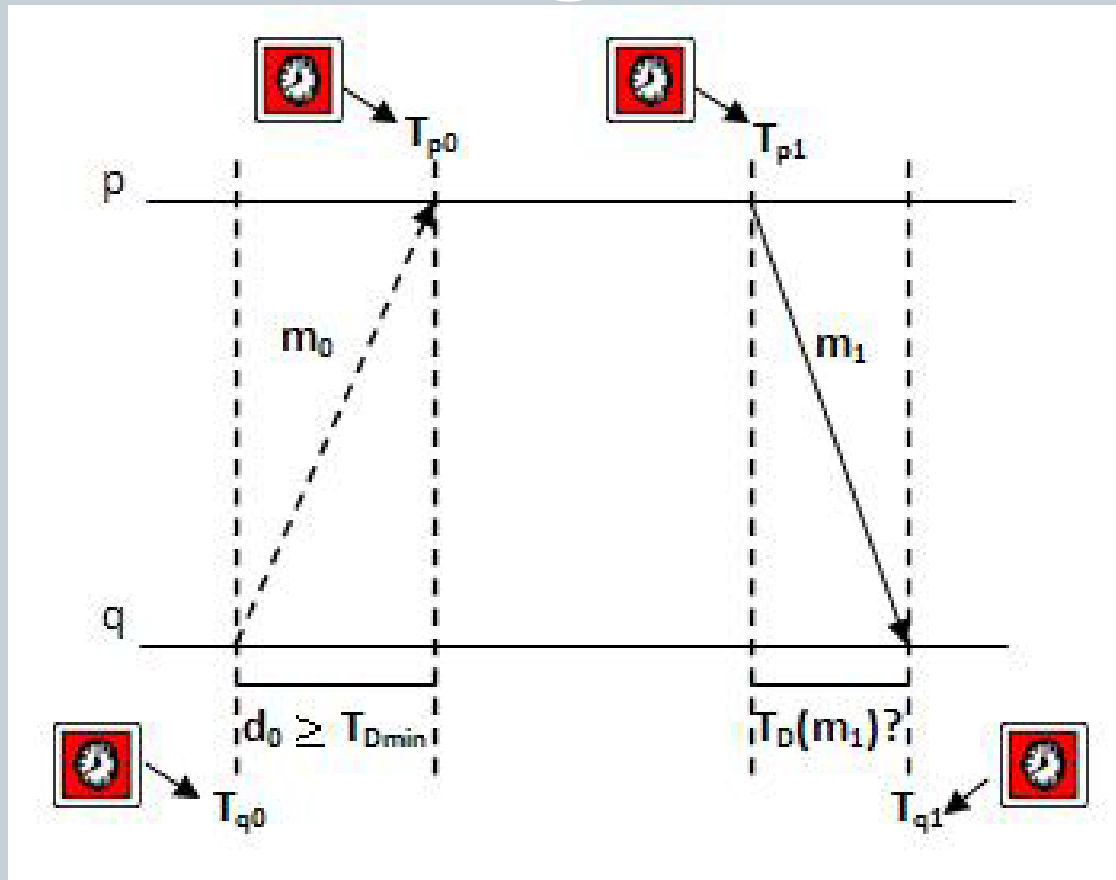
# Medição de durações *round-trip* (cont.)

17

- Pré-requisitos para o uso deste método:
  - Assegurar troca de mensagens frequente entre os *sites* relevantes
  - Assegurar que o *timestamping* das transmissões de mensagens e entregas, também sejam trocados entre os *sites* relevantes

# Medição do atraso de entrega de mensagens

18



$$t_D(m_1) \leq (T_{q1} - T_{q0}) - (T_{p1} - T_{p0}) - T_{Dmin}$$