Modelos Fundamentais de um SD

- Modelo de Falhas/Avarias
- **Modelo de Interação ou Sincronismo**
- Modelo de Segurança

Recordando ....

Modelo de Avarias:
Caracteriza o sistema em termos das falhas/avarias, i.e., dos desvios em relação ao comportamento especificado, que os seus componentes podem apresentar.

Modelo de Sincronismo:
Caracteriza o sistema em termos do comportamento temporal dos seus componentes:

- processos;
- relógios locais;
- canais de comunicação;

Modelo de Sincronismo:

- Síncrono
- Assíncrono

Um sistema diz-se **síncrono** sse:

1 – É conhecido um limite inferior e um limite superior para o tempo que cada processo leva a executar;

2 – É conhecido um limite inferior e um limite superior para o atraso na comunicação entre os seus componentes;

3 - Cada processo tem um relógio local e é conhecido um limite superior para o desvio na sua taxa de incremento.

Modelo de Sincronismo:

Um sistema diz-se **assíncrono** se:

- Nada se assume sobre o comportamento temporal do sistema.

Dilema:

É relativamente fácil resolver problemas com sistemas síncronos, mas é extremamente difícil construir um sistema síncrono.

- Tempo e Relógios

O papel do tempo

- Precisa de ser medido com elevada **precisão**
- Precisa de ser medido de forma **consistente** pelos diversos componentes de um sistema
- Crucial na **ordenação** de eventos

  observadores distintos podem testemunhar eventos por ordens diferentes

O papel do tempo

- Tempo Real?
  - Função monótona contínua e crescente
  - Unidade: segundo

"Actualmente um segundo é a duração de 9.192.631.770 períodos da radiação correspondente à transição entre os 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

O uso do tempo em sistemas distribuídos é feito em dois aspectos:

- Registar 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

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*.

O papel do tempo

Se queremos comparar a duração de vários acontecimentos podemos usar,

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

Durações Distribuídas

- ***Timers* / 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

- **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

- **Relógio físico local (physical clock - pc):** o modo mais comum para fornecer uma fonte de tempo num processo.

  - Equipamento físico, que conta as oscilações que ocorrem num cristal de quartzo a uma dada frequência.
  - Cada oscilação do cristal decrementa o contador de uma unidade.
  - Quando o contador chega a zero, é gerado um *interrupt* e o contador é recarregado com o valor inicial.

  - Cada *interrupt* é designado como um “clock tick”

Relógios locais

- Um relógio num processo correcto, k, implementa uma função **discreta**, monótona crescente, $pc_{k,}$ $q$ue mapeia o tempo real t em tempo de relógio $pc_k(t)$.

- Problemas dos relógios físicos:

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

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

Problemas dos relógios físicos

- A frequência das oscilações varia com a temperatura

- Diferentes taxas de desvio em diferentes computadores

Problemas dos relógios físicos

- **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.

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

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

(Para uma taxa de desvio de 10^{-5} por segundo,

após 60 minutos o erro acumulado pode ser superior a 30 milissegundos

Clock skew / Clock drift

Skew – *é a diferença do valor do tempo lido de dois relógios diferentes.*

Drift – *é a diferença no valor lido de um relógio e o valor do tempo fornecido por um relógio de referência perfeito por unidade de tempo do relógio de referência.*

Ex. *drift de 10 $^{-5}$ segundos / segundo, significa que em cada segundo o relógio tem um desvio de 0,00001 segundos.*

Para que serve um **relógio local**?

- 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

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

Relógios Globais - funcionamento

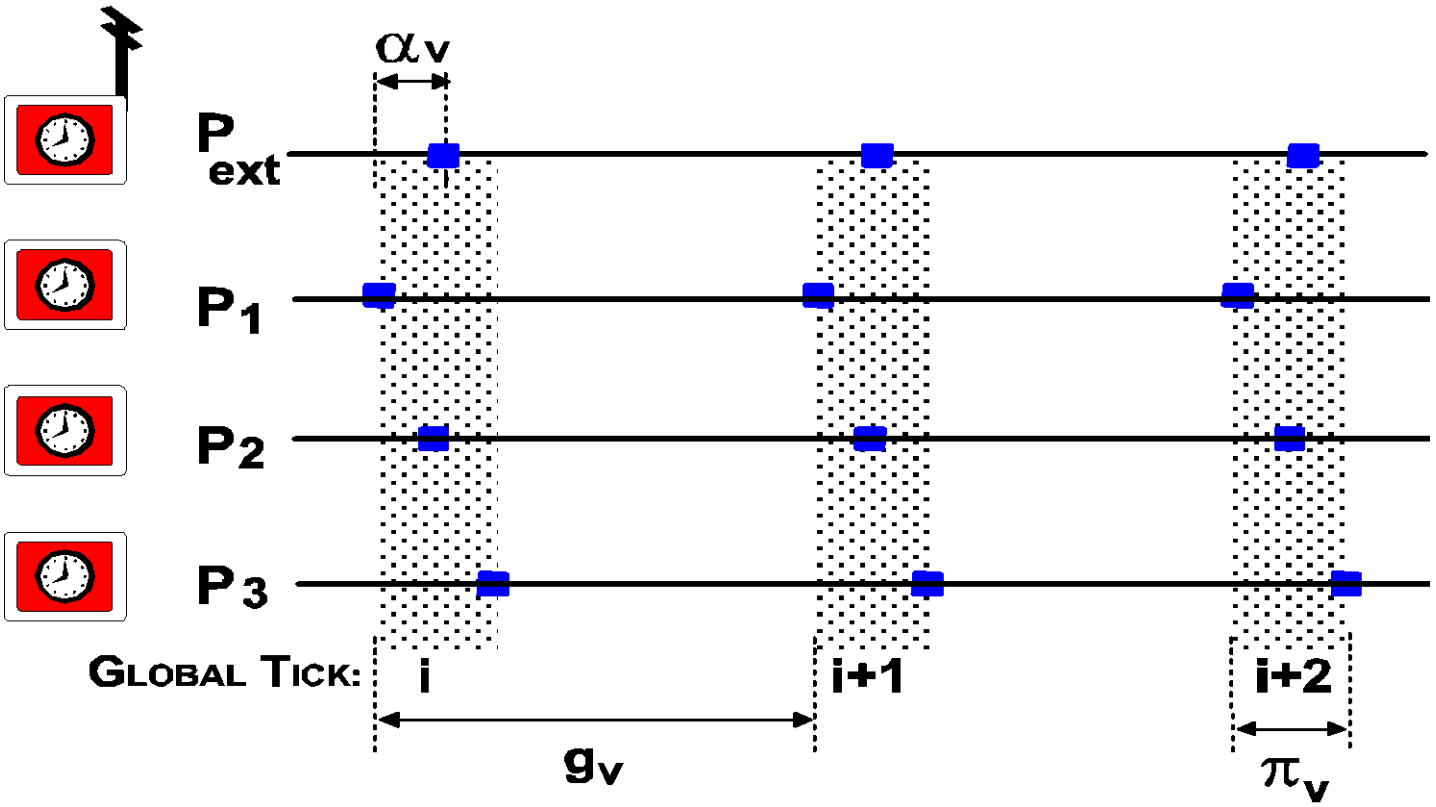
- É criado um relógio virtual vc_p para cada processo *p* a partir do relógio físico.
- É feita a sincronização de todos os relógios locais com o mesmo valor inicial $vc_p(t_{init})$.
- Periodicamente os relógios virtuais são re-sincronizados

Algoritmos de Sincronização => *próxima aula*

Propriedades dos Relógios Globais

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

Propriedades dos Relógios Globais (cont.)



Sincronização interna vs Sincronização externa

- Sincronização interna:

  - relógios têm 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

- Tempo Atómico Internacional (TAI - Temps Atomique International)

  função contínua monótona crescente a uma taxa constante (tempo médio dos relógios atómicos de césio existentes)

- Universal Time, Coordinated (UTC)

  referência de tempo política (correcção do TAI de forma a ajustar o tempo com o dia solar)

- Forma mais simples de obter o UTC: por GPS (Global Position System) – é assegurada uma exactidão, em terra, <= 100ns para os relógios dos receptores de GPS

Medição de durações *round-trip*

- 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

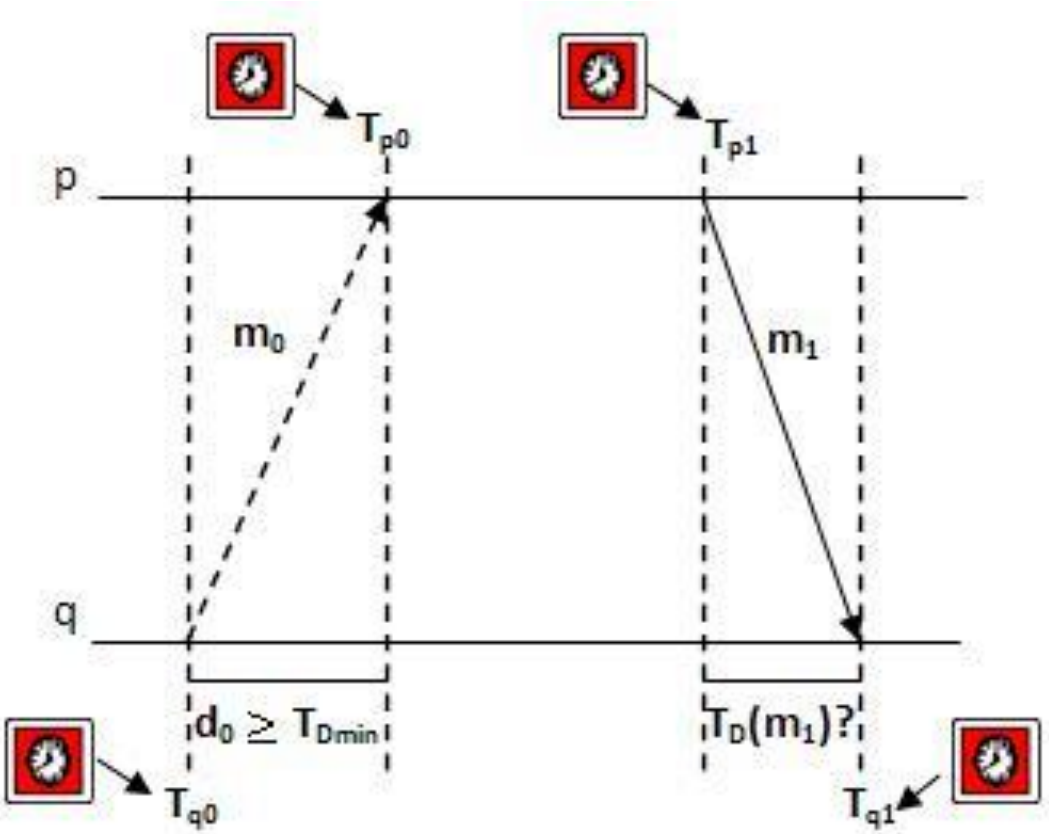
Medição de durações *round-trip* (cont.)

- 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



$$t_D(\mathrm{m}_1) \leq \left(T_{q1} - T_{q0}\right) - \left(T_{p1} - T_{p0}\right) - T_{D\min}$$