Sistemas Distribuídos
Comunicação em Grupo

Disciplina: Sistemas Distribuídos
Prof.: Edmar Roberto Santana de Rezende

Faculdade de Engenharia de Computação
Centro de Ciências Exatas, Ambientais e de Tecnologias
Pontifícia Universidade Católica de Campinas

Comunicação em Grupo

Visão geral

- Comunicação em RPC envolve apenas 2 partes
  1. cliente
  2. servidor
  - Existem circunstâncias onde há necessidade de envio de mensagens para múltiplos processos
  - RPC não suporta comunicação de um emissor para muitos receptores

    → a não ser que seja feita uma chamada de procedimento remoto distinta para cada processo

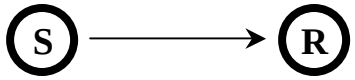
- Necessidade de mecanismos de comunicação alternativos
  - suporte mensagens sendo enviadas para múltiplos receptores em uma única operação

Comunicação em Grupo

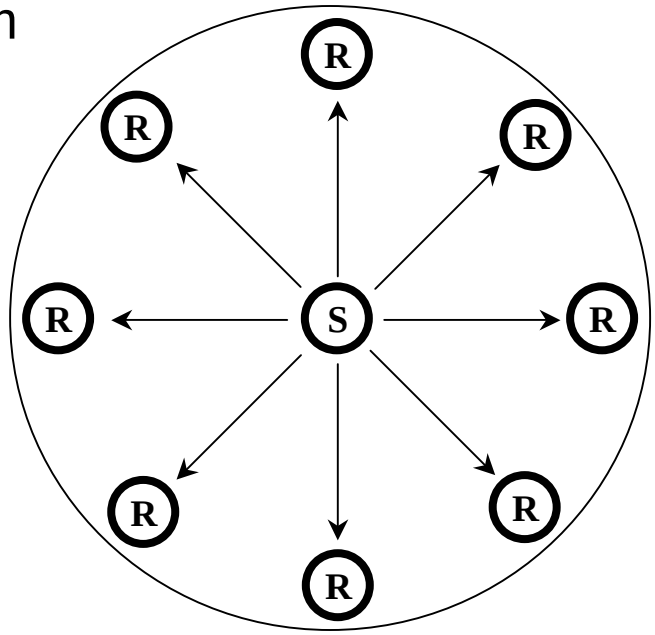
Introdução

- Grupo:
  - uma coleção de processos que agem juntos

→ quando uma mensagem é enviada para o grupo, todos os membros do grupo a recebem



(a) Comunicação "ponto a ponto"



(b) Comunicação "um para muitos"

Comunicação em Grupo

Introdução

- Grupos são dinâmicos:
  - novos grupos podem ser criados e velhos grupos podem ser destruídos
  - processos podem entrar ou abandonar o grupo
  - um mesmo processo pode pertencer a mais de um grupo
  - → são necessários mecanismos para gerenciamento dos grupos e seus membros

- Como a comunicação em grupo pode ser implementada?
  - depende da tecnologia de rede utilizada

Comunicação em Grupo

Introdução

- Multicasting:
  - 1 mensagem enviada a um endereço especial de rede
  → através do qual **múltiplas** máquinas podem ouvir

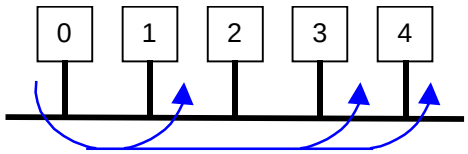
- Broadcasting:
  - 1 mensagem enviada a um endereço especial de rede
  → através do qual **todas** as máquinas podem ouvir

- Unicasting:
  - N mensagens enviadas
  - 1 para cada membro do grupo

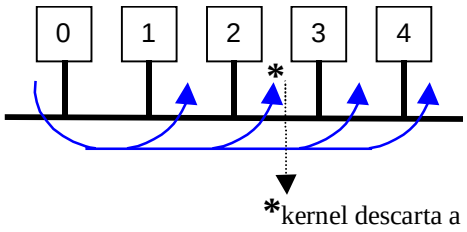
Comunicação em Grupo

Introdução

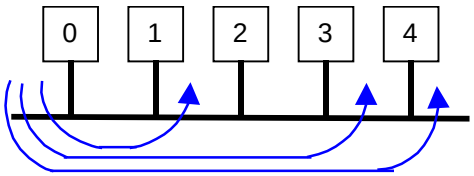
- Formas de implementar a comunicação em grupo:



(a) Multicast



(b) Broadcast



(c) Unicast

Comunicação em Grupo

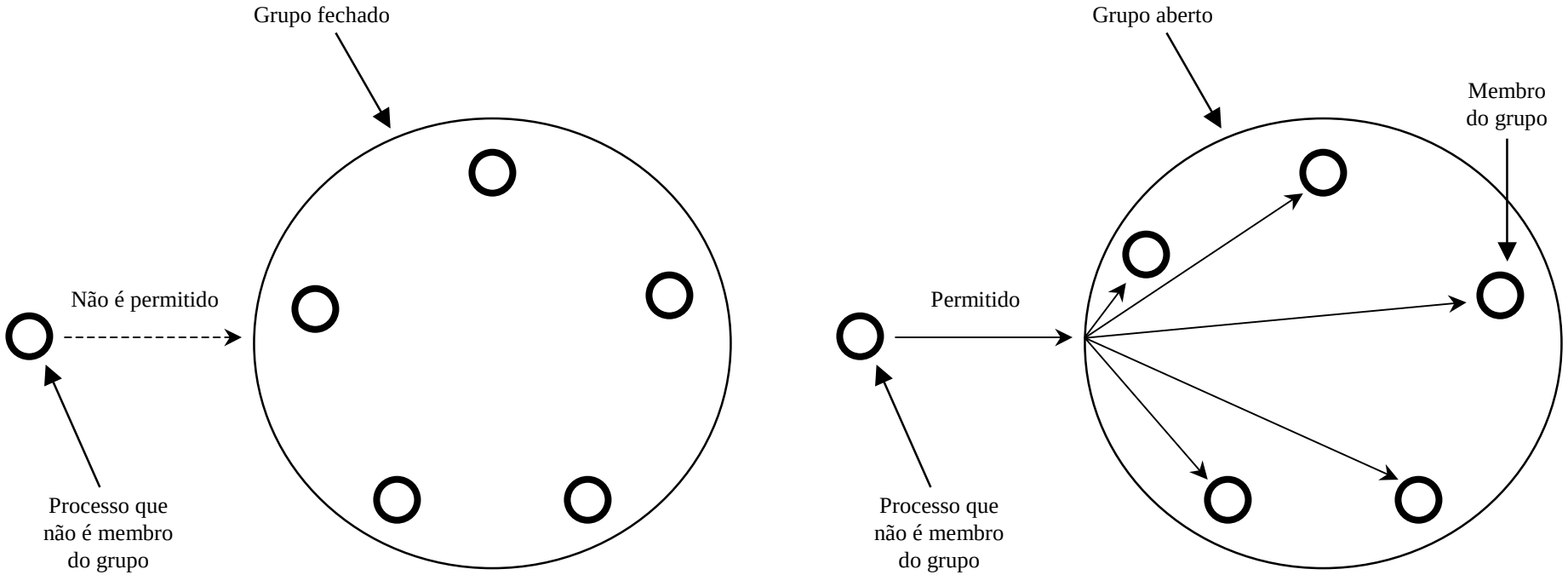
Questões de Projeto

- Grupos abertos x Grupos fechados
  - essa diferenciação é definida em função de quem pode ou não se comunicar com os elementos do grupo
  - Grupos abertos
    - qualquer processo no sistema pode enviar mensagem para o grupo
    - abordagem usada tipicamente em casos de replicação de serviços
  - Grupos fechados
    - apenas os membros do grupo podem mandar mensagem para o grupo
    - processos fora do grupo não podem enviar mensagens ao grupo como um todo, apenas aos membros individualmente
    - abordagem usada tipicamente em processamento paralelo

Comunicação em Grupo

Questões de Projeto

- Grupos abertos x Grupos fechados



Comunicação em Grupo

Questões de Projeto

- Grupos igualitários x Grupos hierárquicos
  - essa classificação indica a estrutura interna do grupo

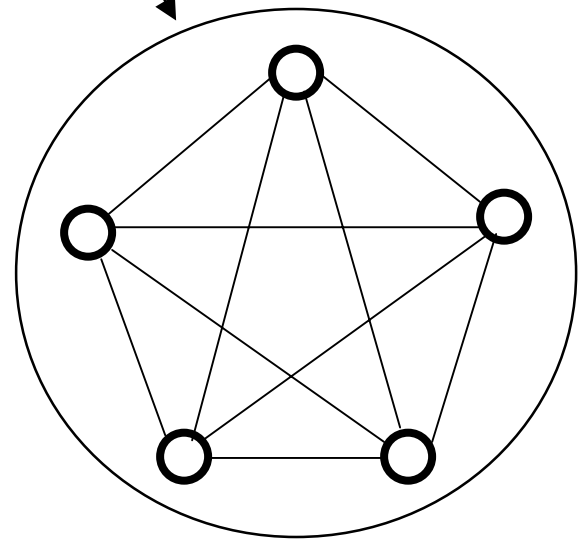
  - Grupos igualitários (pares)
    - todos os processos são iguais e as decisões são tomadas coletivamente
    - ☺ grupo simétrico
    - ☺ ausência de ponto único de falhas
    - ☹ a tomada de decisão é bem mais complexa

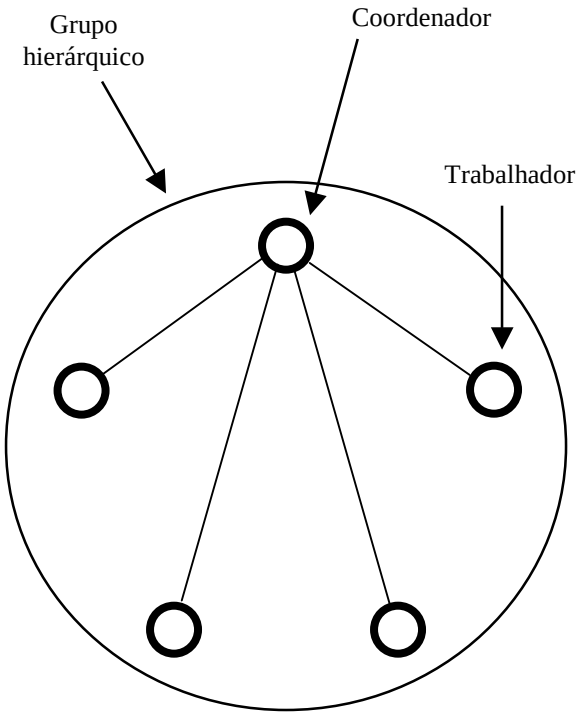
  - Grupos hierárquicos
    - existe algum tipo de hierarquia entre os membros
    - organização mais simples consiste em:
      - um processo coordenador
      - outros processos denominados trabalhadores
    - quando uma requisição é feita:
      - o coordenador decide qual trabalhador é o mais apropriado para realizar a tarefa, ou define que todos devem realizar a tarefa
    - ☹ ponto único de falha
    - ☺ simplificação da tomada de decisão

Comunicação em Grupo

Questões de Projeto

- Grupos igualitários x Grupos hierárquicos





Comunicação em Grupo

Questões de Projeto

- Controle dos membros do grupo
  - Grupos são dinâmicos
    - → é preciso realizar de alguma forma o controle de quais são os membros do grupo, bem como permitir a inclusão e exclusão de membros
  - As duas principais formas de fazer essa **gerência** são através de um:
    1. servidor de grupo
    2. controle de forma distribuída

Comunicação em Grupo

Questões de Projeto

- Controle dos membros do grupo
  - *Servidor de Grupo*:
    - todas as requisições devem ser feitas ao servidor
    - o servidor deve manter uma base de dados completa de todos os grupos e seu conjunto exato de membros
    - ☺ eficiente
    - ☺ direto
    - ☺ fácil de implementar
    - ☹ ponto único de falha

Comunicação em Grupo

Questões de Projeto

- Controle dos membros do grupo
  - *Controle de forma distribuída*:
    - cada novo membro deve enviar mensagem a todos os membros atuais anunciando a sua presença
      - → ao sair, um membro deve avisar a todos os outros membros
    - ☺ facilita uma implementação tolerante a falhas
    - ☹ implicações:
      - quando um membro falha ele deixa de pertencer ao grupo
        - → não há mensagens de aviso!
      - a entrada e a saída do grupo devem ser sincronizadas com a troca de mensagens

Comunicação em Grupo

Questões de Projeto

- Endereçamento do grupo
  - é necessário haver uma forma de indicar para qual grupo será enviada a mensagem
  - Solução:
    1. dar a cada grupo um endereço único
       - grupo associado a um endereço multicast

         » mensagens serão enviadas somente às máquinas pertencentes ao grupo
       - grupo associado a um endereço broadcast

         » mensagens são enviadas a todas as máquinas

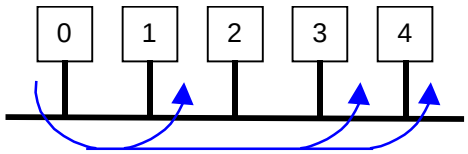
         → se nenhum processo na máquina for membro do grupo a mensagem é descartada
       - grupo associado a uma lista de endereços unicast

         » o kernel envia uma mensagem para cada endereço da lista

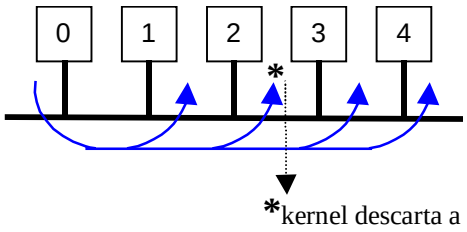
Comunicação em Grupo

Questões de Projeto

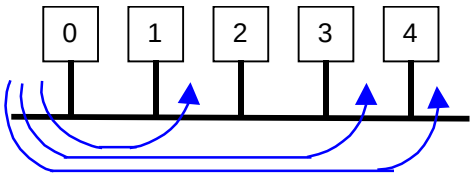
- Endereçamento do grupo



1 Multicast



1 Broadcast



3 Unicasts

Comunicação em Grupo

Questões de Projeto

- Endereçamento do grupo
  - Solução:
    2. o emissor deve fornecer uma lista de todos os destinatários
       - uma lista de endereços é especificada na chamada *send*
       - ☹ força os processos a conhecerem todos os membros do grupo
         → não é transparente
       - ☹ quando alguém entra ou sai do grupo
         → o processo deve atualizar sua lista de membros

    3. uso de predicados (expressões booleanas)
       - mensagem enviada a todos os membros do grupo (ou possivelmente todo o sistema) usando um dos métodos anteriores
       - somente quando o predicado for avaliado com verdadeiro o membro irá aceitar a mensagem

Comunicação em Grupo

Questões de Projeto

- Primitivas de comunicação com o grupo
  - as primitivas de comunicação ponto a ponto e em grupo devem ser definidas em um mesmo conjunto
  - RPC
    - o envio de mensagens para um grupo não pode ser modelado como uma chamada de procedimento remoto
    - o cliente envia 1 mensagem e o servidor envia 1 resposta
    - com a comunicação em grupo existem potencialmente N respostas diferentes
      - → como lidar com N respostas?
  - Consequência:
    - necessidade de chamadas explícitas *send* e *receive*

Comunicação em Grupo

Questões de Projeto

- Primitivas de comunicação com o grupo
  - *receive*
    - aguarda por mensagens tanto “ponto a ponto” quanto do grupo
  - *send*
    - endereço de processo: mensagem enviada para um único processo
    - endereço do grupo (ou uma lista de endereços): mensagem enviada para todos os membros do grupo
  - Bloqueio, bufferização e confiabilidade
    - seguem os mesmo princípios da comunicação “ponto a ponto”
    - geralmente tais decisões são fixas, e não por mensagem
  - Comunicação “ponto a ponto” e em grupo possuem diferentes propósitos
    - introduzir novas primitivas *group_send* e *group_receive*

Comunicação em Grupo

Questões de Projeto

- Atomicidade
  - propriedade de entregar “tudo ou nada”
  - quando uma mensagem é enviada para o grupo
    - ela deverá chegar corretamente para todos os membros ou para nenhum deles
    - situações onde alguns membros recebem a mensagem e outros não, não são permitidas
  - torna a programação mais fácil
    - quando um processo envia uma mensagem para o grupo não precisa se preocupar com o fato de alguém não ter recebido a mensagem
  - falhas na entrega são reportadas ao emissor
    - pode realizar as ações necessárias para recuperação

Comunicação em Grupo

Questões de Projeto

- Atomicidade
  - Implementação não é simples
    - a única forma de garantir a entrega da mensagem para todos os destinos é através do envio de mensagens de reconhecimento
    - situações onde alguns membros recebem a mensagem e outros não, não são permitidas
  - Tolerância a falhas
    - é essencial que a atomicidade seja mantida mesmo na presença de falhas

    Ex:

    1. emissor envia uma mensagem para todos os membros do grupo
    2. um processo do grupo não a recebe
    3. o emissor falha

    → alguns membros do grupo receberam a mensagem e outros não

    → não há mais um processos para realizar a retransmissão

Comunicação em Grupo

Questões de Projeto

- Atomicidade
  - Solução:
    1. emissor envia uma mensagem para todos os membros do grupo
       1.a) temporizadores são configurados
       1.b) retransmissões podem ser feitas quando necessário
    2. quando um processo recebe a mensagem pela 1ª vez
       2.a) envia a mensagem para todos os membros
       2.b) com o uso de temporizadores e retransmissões se necessário
    2'. quando um processo recebe a mensagem já recebida
       2'.a) descarta a mensagem

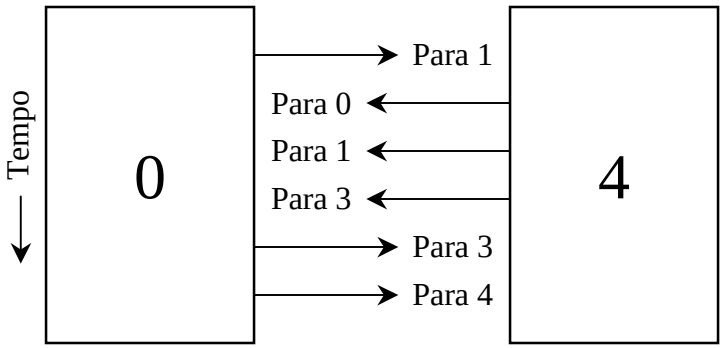
  - não importa quantos processos falhem ou quantos pacotes são perdidos
    → todos os processos sobreviventes receberão a mensagem

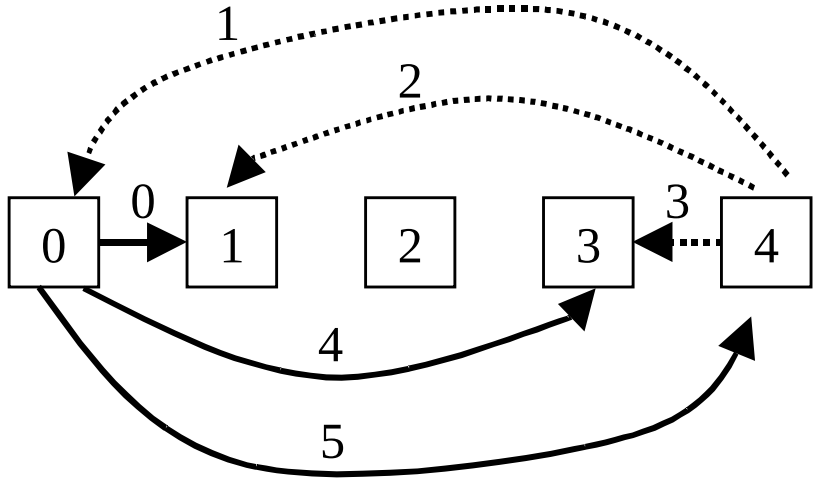
Comunicação em Grupo

Questões de Projeto

- Ordenação de mensagens
  - torna a comunicação em grupo mais simples
  - Problemas:
    - processos 0 e 4 enviam mensagem para o grupo ao mesmo tempo

    → a ordem em que as mensagens chegam não pode ser determinada





Comunicação em Grupo

Questões de Projeto

- Ordenação de mensagens
  - se os processos 0 e 4 estiverem utilizando o mesmo registro em um banco de dados

    → 1 e 3 possuirão valores finais diferentes

  - Soluções:
    - ordenação em tempo global
      - todos os processos do grupo recebem as mensagens exatamente na mesma ordem

        → mesma ordem de envio das mensagens

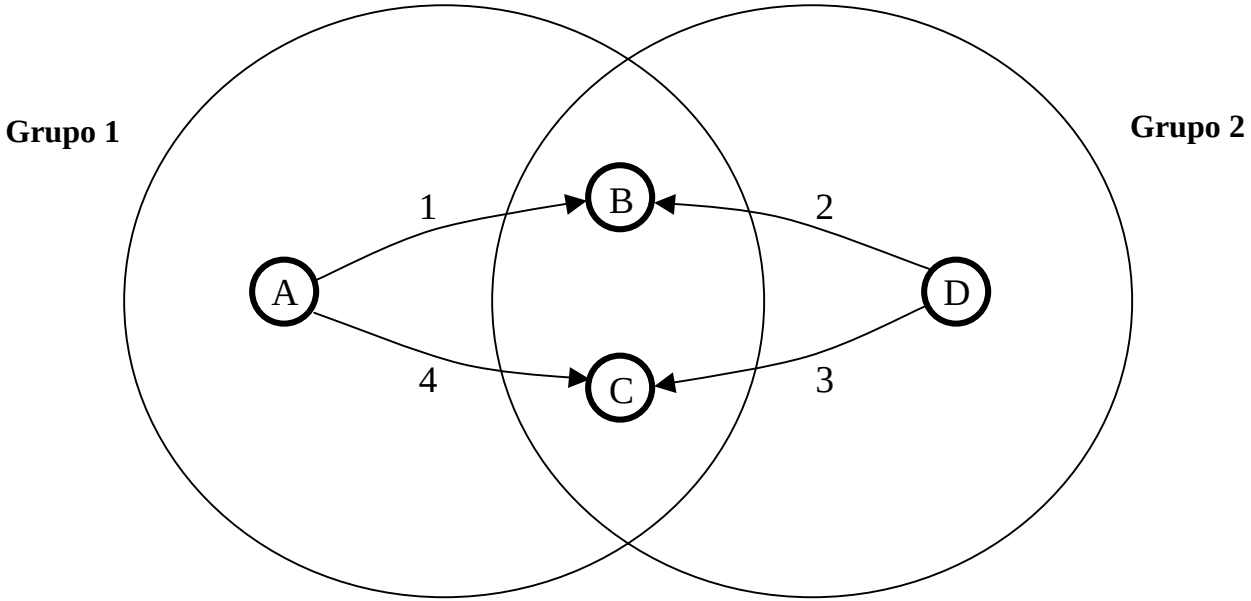
      ☹ não é fácil de se implementar
    - ordenação em tempo consistente
      - todos os processos do grupo recebem a mensagens na mesma ordem

        → pode não ser a mesma ordem de envio das mensagens

Comunicação em Grupo

Questões de Projeto

- Sobreposição de grupos
  - processos podem pertercer a mais de um grupo
    → é possível que surjam inconsistências



Comunicação em Grupo

Questões de Projeto

- Escalabilidade
  - muitos algoritmos funcionam bem para grupos com poucos membros

    → o que acontece quando se tem milhares de membros por grupo? Ou milhares de grupos?

    → o que acontece quando o sistema toma proporções maiores do que uma rede local? Ou grupos se espalham por continentes?
  - Problemas:
    - multicast

      → necessidade de algoritmos mais sofisticados mantendo rastros
    - o uso de gateways dificulta uma ordenação por tempo global absoluto

      → mais de um pacote ao mesmo tempo na rede
    - algoritmos escaláveis

      → aumento da complexidade