

SSC150 – Sistemas Computacionais Distribuídos

Comunicação em Sistemas Distribuídos
Comunicação em grupo

6ª aula
15/04/10

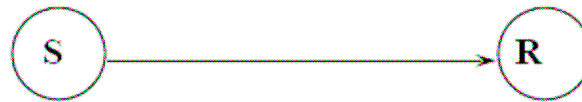
Profa. Sarita Mazzini Bruschi
sarita@icmc.usp.br

Slides baseados no material de:
Prof. Rodrigo Mello (USP / ICMC)
Prof. Edmilson Marmo Moreira (UNIFEI / IESTI)

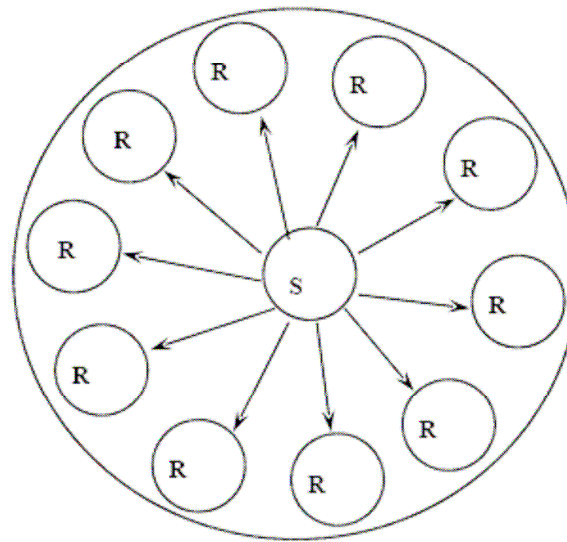
Comunicação em grupo

- A comunicação em RPC envolve somente duas partes: o cliente e o servidor
- Em algumas circunstancias é necessário envolver múltiplos processos
- Exemplo:
 - Um grupo de servidores de arquivos cooperando para fornecer um sistema único e tolerante a falhas. É necessário enviar uma mensagem para todos os servidores
- Comunicação em grupo é um mecanismo de comunicação alternativo no qual a mensagem pode ser enviada para múltiplos receptores

Comunicação em grupo



Comunicação ponto-a-ponto



Comunicação um-para-vários

Comunicação em grupo

- Os grupos são dinâmicos
- Novos grupos podem ser criados e grupos existentes podem ser eliminados
- Um processo pode entrar para um grupo ou deixá-lo
- Um processo pode ser membro de diversos grupos simultaneamente

Comunicação em grupo

Implementação

■ Multicasting

- ❑ Endereço especial que múltiplas máquinas podem receber
- ❑ Implementação direta: basta atribuir a cada grupo um endereço multicasting diferente

■ Broadcasting

- ❑ Os pacotes são enviados para todas as máquinas
- ❑ Menos eficiente que multicasting
- ❑ Também necessita somente um pacote para atingir a todos os membros do grupo

■ Unicasting

- ❑ Transmissão separada de pacotes para cada membro do grupo (n membros \rightarrow n pacotes)

Comunicação em grupo

Grupo Fechado e Grupo Aberto

■ Grupo Fechado

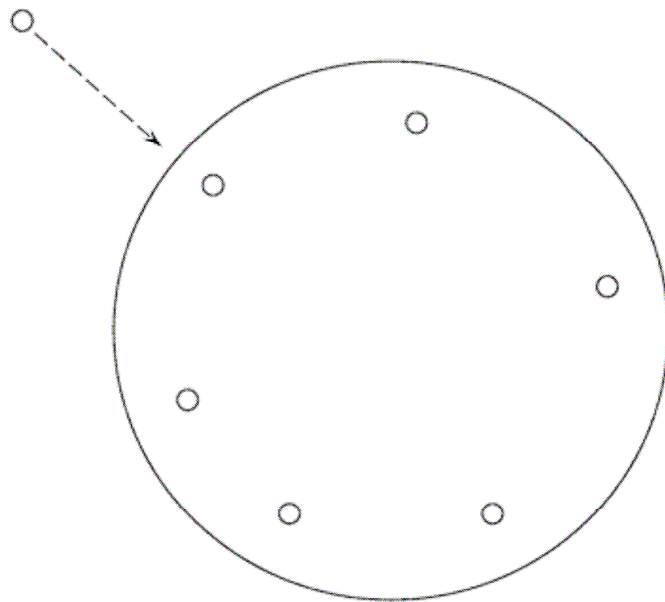
- ❑ Somente os membros do grupo podem enviar mensagens para o grupo
- ❑ Normalmente usado para processamento paralelo

■ Grupo Aberto

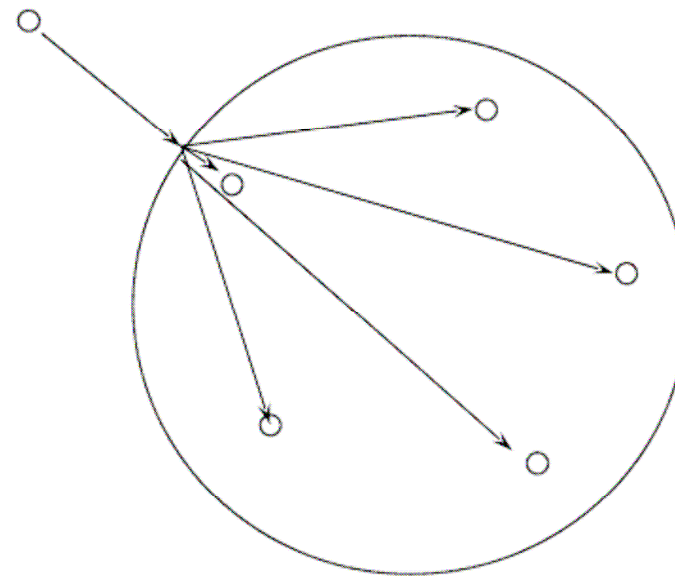
- ❑ Qualquer processo pode enviar uma mensagem para qualquer um do grupo
- ❑ Exemplo: um servidor de arquivo replicado

Comunicação em grupo

Grupo Fechado e Grupo Aberto



Grupo Fechado



Grupo Aberto

Comunicação em grupo

Grupo Hierárquico e Grupo “de iguais”

■ “De iguais”

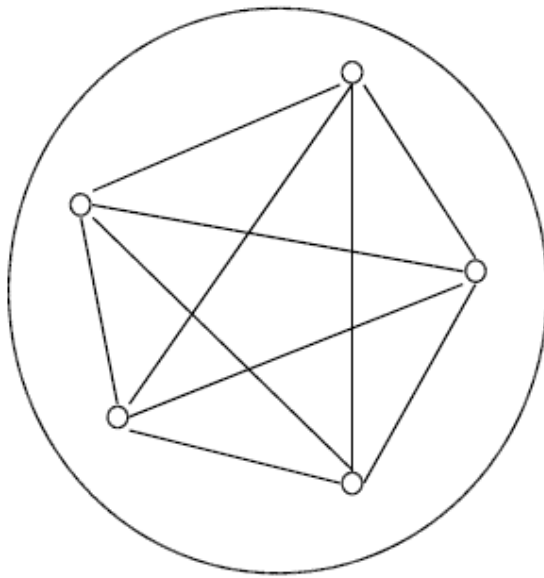
- ❑ Todos são iguais e não existe um coordenador
- ❑ Vantagem: é simétrico e não possui ponto de falha
- ❑ Desvantagem: Decisão mais complicada

■ Hierárquico:

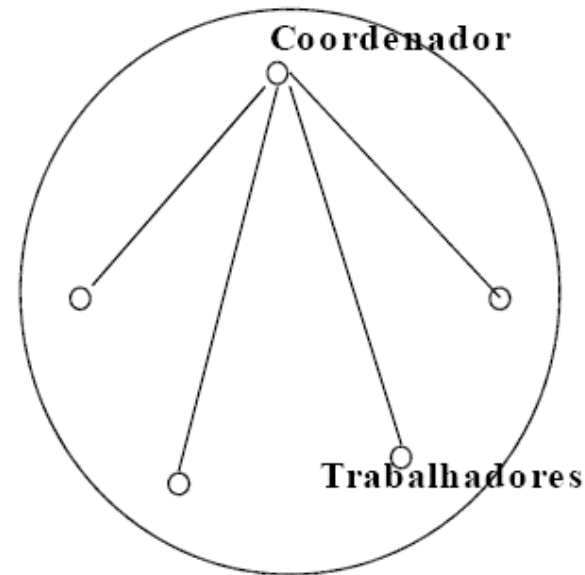
- ❑ Existe alguma hierarquia
- ❑ Vantagem: decisões mais rápidas
- ❑ Desvantagem: a perda do coordenador para o grupo todo

Comunicação em grupo

Grupo Hierárquico e Grupo “de iguais”



Grupo “de Iguais”



Grupo Hierárquico

Comunicação em grupo

Servidor de Grupo

- Função: Criação e eliminação de grupos e permissão para processos aderirem ou abandonarem os grupos
- Um único servidor monta uma base de dados com informações dos grupos, o que pode ser um ponto crítico de falha
 - Outra forma é fazer o gerenciamento de forma distribuída
- Quando um processo deixa o grupo, envia uma mensagem de *goodbye* para todos os outros membros
- Problema: se um membro falha, ele deixa o grupo sem o *goodbye*. Os outros membros têm que descobrir isso explicitamente e removê-lo do grupo

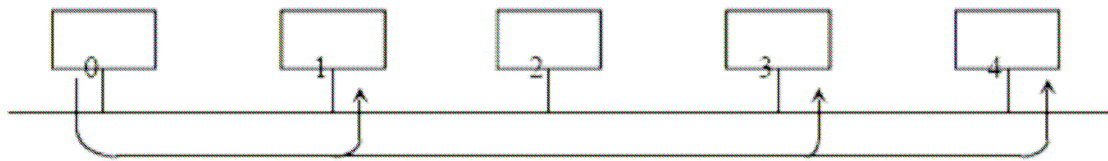
Comunicação em grupo

Endereçamento

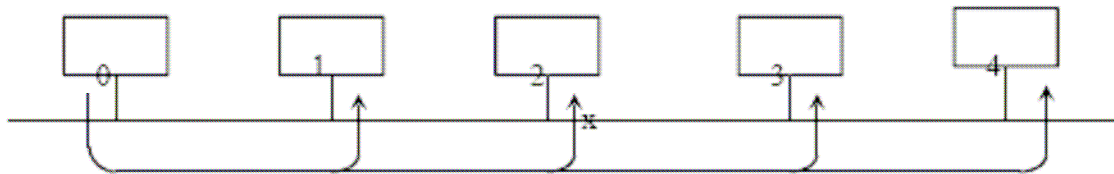
- Da mesma maneira que os processos, os grupos também precisam ser endereçados
- Unicast: processo que envia a mensagem fornece uma lista de todos os endereços destinos (não transparente)
- Endereçamento com predicado:
 - cada mensagem é enviada para todo o grupo ou para todo o sistema.
 - cada mensagem contém uma expressão booleana para ser avaliada
 - se o resultado for verdadeiro, a mensagem é aceita, caso contrário, é descartada

Comunicação em grupo

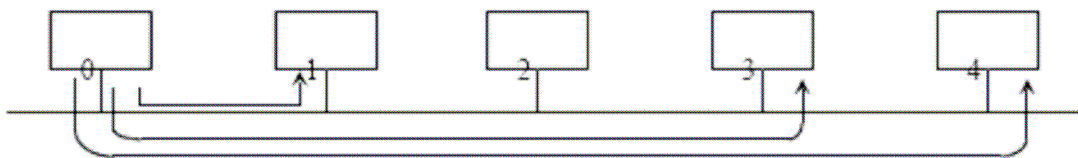
Endereçamento



Multicast



Broadcast



Unicast

Comunicação em grupo

Primitivas Send e Receive

- É difícil juntar RPC e comunicação em grupo
 - Enquanto no RPC existe uma única requisição e uma única resposta, na comunicação em grupo serão várias respostas para cada requisição
 - Melhor utilizar explicitamente as primitivas *send* e *receive* (uma única via de comunicação) ao invés de *request/reply* (duas vias de comunicação)
- Pode haver:
 - `group_send` e `group_receive`
 - Um processo indica se quer comunicação ponto-a-ponto ou de grupo

Comunicação em grupo

Atomicidade

- Propriedade relevante da comunicação em grupo:
 - Evitar que um participante do grupo não tenha recebido a mensagem
- Quando uma mensagem é enviada para um grupo, ela chega corretamente para todos os membros do grupo ou não chega para nenhum

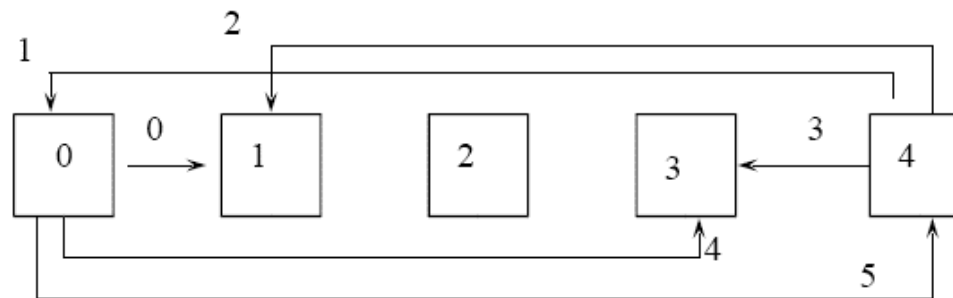
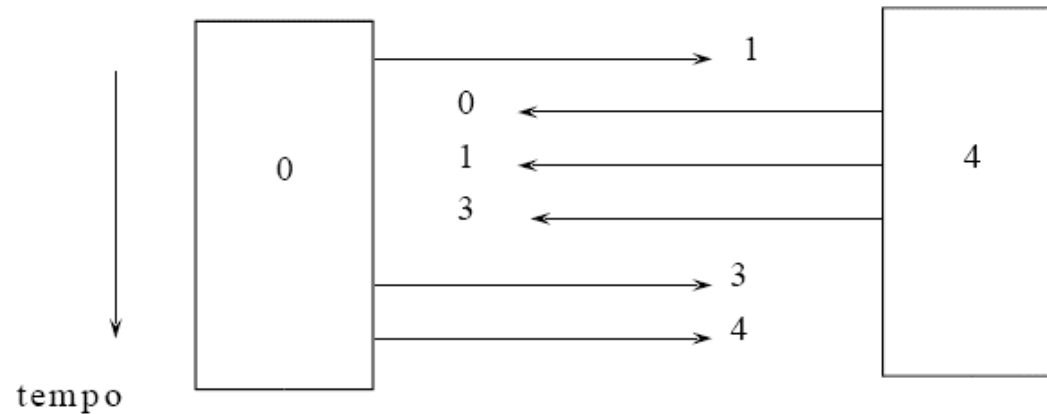
Comunicação em grupo

Atomicidade

- Uma maneira de ter certeza que todos os destinatários receberam a mensagem é implementar o envio de ACK para cada mensagem recebida
 - Sem falhas, esse método funciona
 - Com falhas:
 - Processo que envia mensagens inicia timers e envia retransmissões quando necessário
 - Quando um processo recebe uma mensagem e não a havia visto ainda, envia mensagem para todos os membros do grupo

Comunicação em grupo

Ordenando mensagens



Processos 0 e 4 enviando mensagens que podem chegar em instantes distintos
Podem causar problemas: exemplo atualizando registros em banco de dados

Comunicação em grupo

Ordenando mensagens

■ **Global Time Ordering**

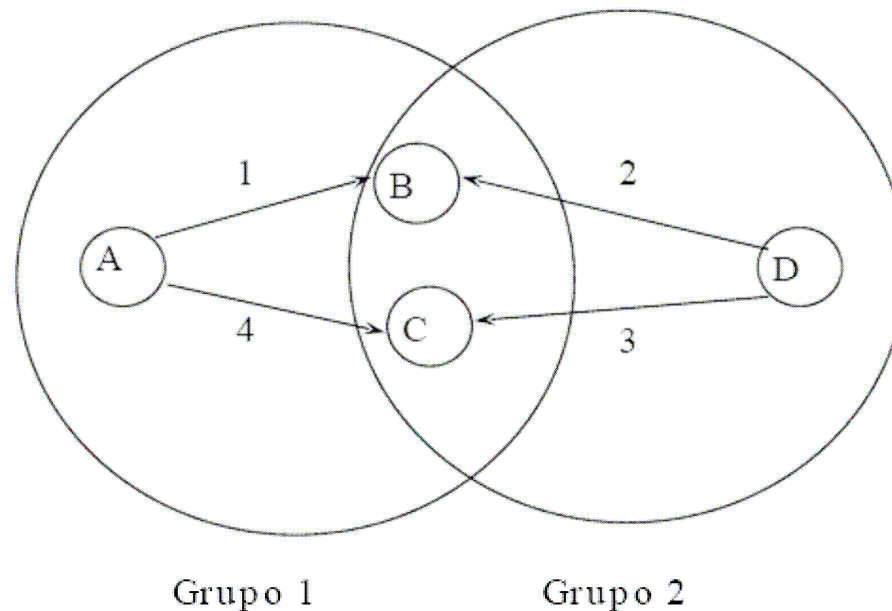
- todas as mensagens chegam na ordem exata em que foram enviadas
- não é fácil de ser implementado → se mensagem A foi gerada antes de B, então A chega a todos os processos antes de B

■ **Consistent Time Ordering**

- se duas mensagens são enviadas em tempos próximos, o sistema pega uma delas como sendo a “primeira” e envia a todos os membros do grupo segundo esta ordem
- é garantido que as mensagens cheguem a todos os membros do grupo na mesma ordem, que pode não ser a ordem real

Comunicação em grupo

Ordenando mensagens em Overlapping Groups



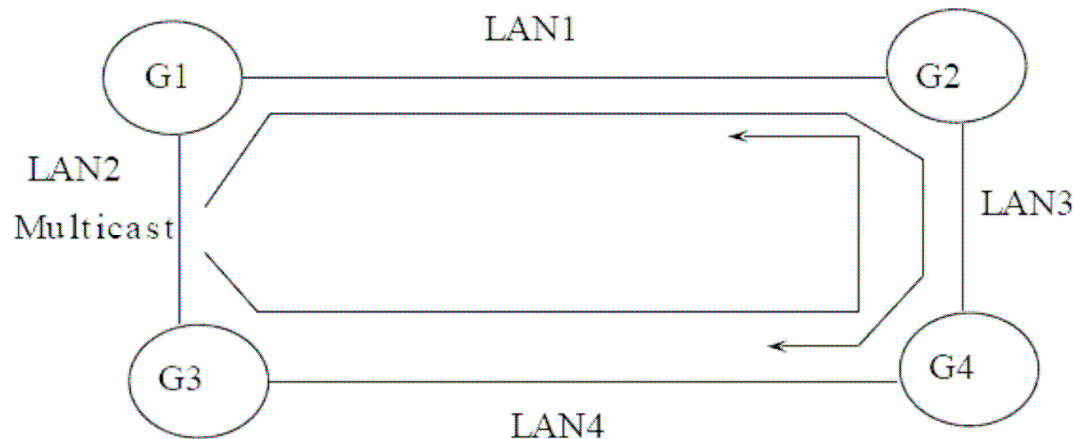
Mesmo havendo ordenação de tempo global no grupo, essa não existe entre grupos !!!

Mesmo usando GTO, pode haver mensagens chegando em ordem diferente

Comunicação em Grupo

Escalabilidade

- O que acontece com muitos membros por grupo? Com muitos grupos? Em redes distintas?



- Multicast é mais complicado
 - mensagem multicast enviada na rede 2 pode passar pelos Gateways e chegar à outras redes, sendo replicadas.