

Tolerância a Falhas



falhas em sistemas distribuídos

- Lamport: “A distributed system is a system where I can’t get any work done if a machine I’ve never heard of crashes.”
 - sistemas distribuídos e falhas parciais
 - tolerância a falhas



Erros, falhas, etc

- problemas inevitáveis: *falhas (faults)*
 - máquinas quebradas, desconexões, erros no software
- erros (failures) -> consequências dessas falhas
- nomenclaturas variam
 - mas temos que manter consistente a idéia de *tolerância a falhas*
 - tolerância a falhas: evitar que falhas se transformem em erros
- classificação de falhas



Tipos de falhas

- modelos baseados em comportamento de servidores
 - omissão
 - temporização
 - falhas arbitrárias
 - ou bizantinas



falhas de omissão

- fail-stop
 - processo “cai” e isso é detectável por parceiros
- crash
 - processo “cai” e parceiros podem não detectar
 - relação com tempo de comunicação
- omissão
 - processo envia mensagem mas ela não é recebida do outro lado



falhas de temporização

- relacionadas com restrições temporais:
 - relógio físico tem desvio superior ao permitido
 - transmissão de mensagem demora tempo demais
 - ...
- relação com modelos *síncronos*



falhas arbitrarias

- respostas podem ocorrer ou não
- conteúdo pode ser correto ou não
 - difícil detecção!
 - também chamadas de falhas bizantinas



Tolerância a falhas & dependabilidade

- uso de “tolerância a falhas” por vezes considerado enganador
 - dependabilidade: idéia de que se pode confiar no sistema (apesar de possíveis erros)

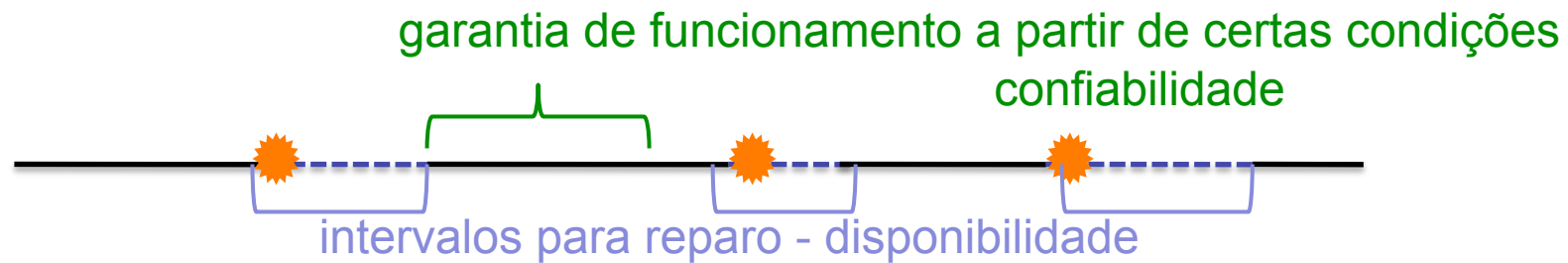


Dependabilidade - atributos

- Confiabilidade (*reliability*)
 - capacidade de atender a especificação, dentro de condições definidas, durante certo período de funcionamento e condicionado a estar operacional no início do período
- Disponibilidade (*availability*)
 - probabilidade do sistema estar operacional num instante de tempo determinado; alternância de períodos de funcionamento e reparo
- Segurança (*safety*)
 - probabilidade do sistema ou estar operacional e executar sua função corretamente ou descontinuar suas funções de forma a não provocar dano a outros sistema ou pessoas que dele dependam
- Segurança (*security*)
 - proteção contra falhas maliciosas, visando privacidade, autenticidade, integridade e irrepudiabilidade dos dados



confiabilidade e disponibilidade



- medidas relacionadas:
 - MTTR – tempo médio de reparo
 - MTBF – tempo médio entre falhas



Técnicas de Dependabilidade

- detecção da falha
- ..., localização, confinamento
- reconfiguração
- recuperação de erro

ou

- mascaramento



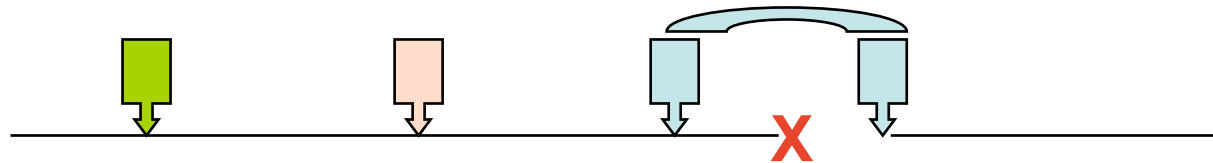
recuperação

- passa o sistema para um estado correto
 - forward recovery
 - backward recovery
 - relação com logging e checkpoints



recuperação de falhas com checkpoints

- processos gravam infos de estado de tempos em tempos para poderem recuperar estado



- gravação em log estável ou em réplicas
 - mensagens enviadas ou recebidas
 - globais
 - em alguns casos pilha completa



checkpoints distribuídos

- os checkpoints dos diversos processos em uma aplicação distribuída não podem ocorrer de forma independente
 - cortes consistentes e checkpoints coordenados



mascamamento

- técnica básica é a redundância
- classes de redundância:
 - redundância de informação
 - códigos como Hamming
 - redundância temporal
 - operações repetidas (por exemplo, em transações)
 - redundância física
 - **processos, dados** ou hardware



Replicação

- tolerância a falhas
 - correção e disponibilidade
- desempenho
 - ... é ou não um caso de distribuição instrínscica?

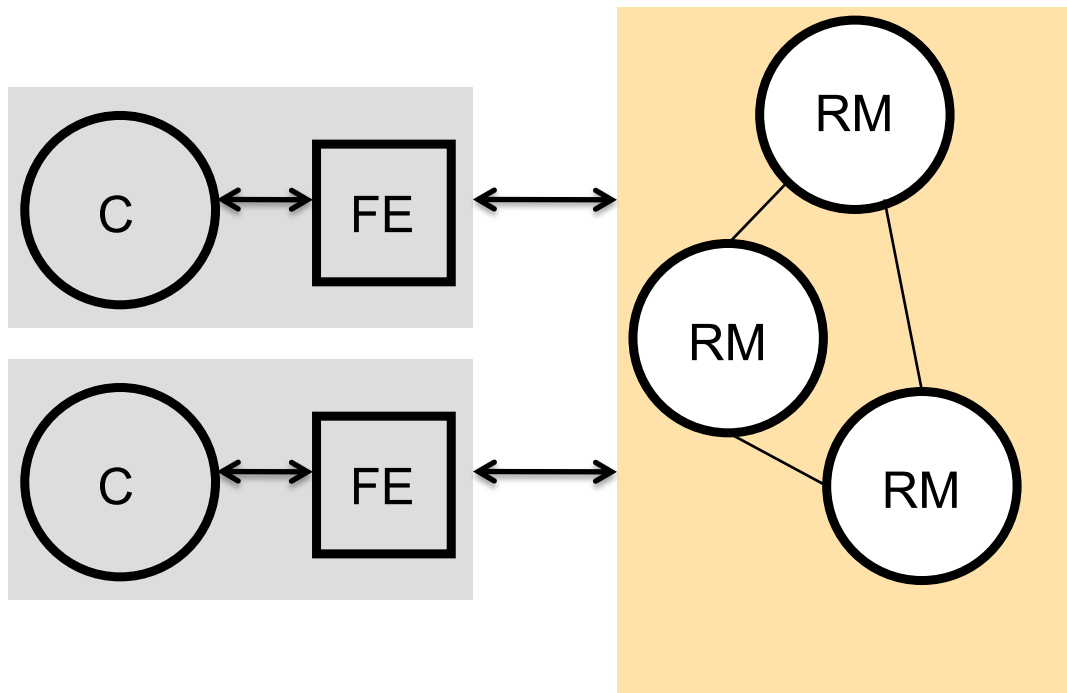


modelo de sistema

- falhas do tipo *crash*
- partições na rede não ocorrem
- sistema composto por *gerentes de réplicas* (ou servidores)
- cada gerente de réplica sabe fazer recuperações
- conjunto de réplicas pode ser estático ou dinâmico



replicação



- transparência
- consistência



processamento de requisições

1. envio da msg pelo front-end
 - para uma réplica ou para todas
2. coordenação
 - gerentes se coordenam para fazer a entrega
3. execução
 - réplicas executam a requisição
4. acordo
 - consenso sobre o efeito da requisição
5. resposta
 - uma ou mais réplicas respondem ao front-end



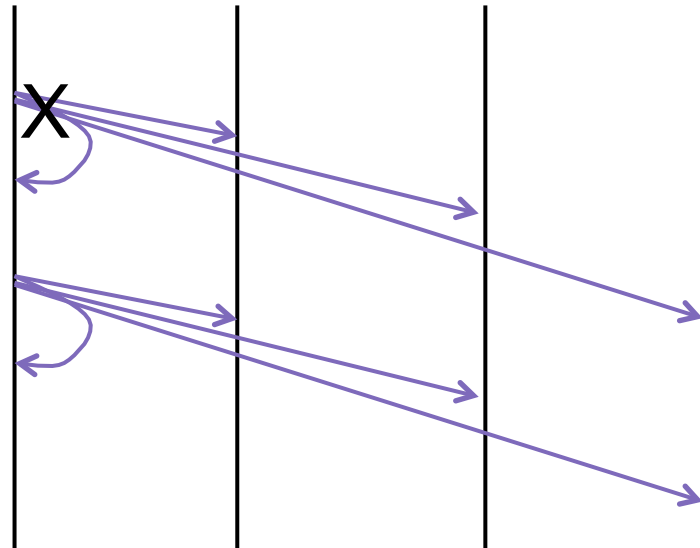
Grupos de processos

- serviços de envio
 - envio atômico
 - envios ordenados: fifo, causal e total
- serviços de controle de participantes
- servidor de grupo ou membership server
- saídas do grupo também podem ocorrer por falhas

- gerentes de réplicas responsáveis por implementação de filas de entrega, etc



ordenação com falhas



- multicast confiável
- multicast atômico: confiável + ordem total



ordenação causal ou total

- como visto antes



grupos estáticos e dinâmicos

- para tolerância a falhas, é importante levar em consideração a possibilidade de entradas e saídas



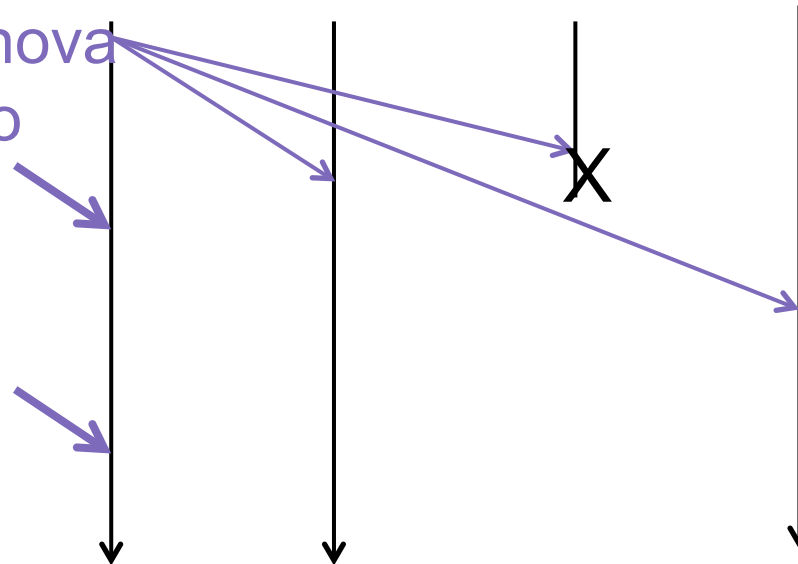
serviços de gerência de grupos

- interface de acesso:
 - criação e destruição de grupos
 - retirada e adesão de processos a grupos
- serviço de detecção de falhas
- notificação:
 - serviço avisa membros do grupos sobre entradas e saídas (programadas ou não)
 - falamos em *visões* do grupo
 - *view-synchronous group communication*



controle de grupo

avisos sobre nova
visão do grupo



novo
processo

- necessidade de manter as visões consistentes com outras atividades

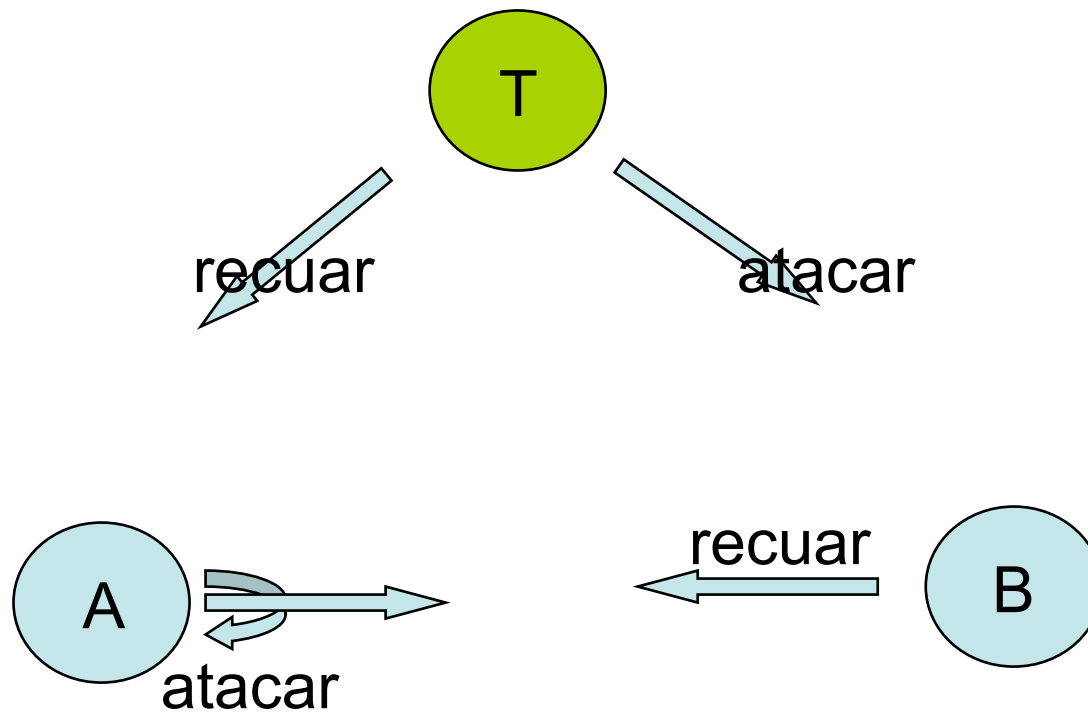


consenso em sistemas distribuídos

- em várias situações, os processos de um grupo têm que chegar a um valor comum
 - ordenação total
 - coordenação de atividades
 - valor a ser respondido em uma consulta
 - ...
- em sistemas com falhas bizantinas...
 - algoritmos de consenso
 - pelo menos $2/3$ dos processos devem estar corretos



problema dos generais bizantinos

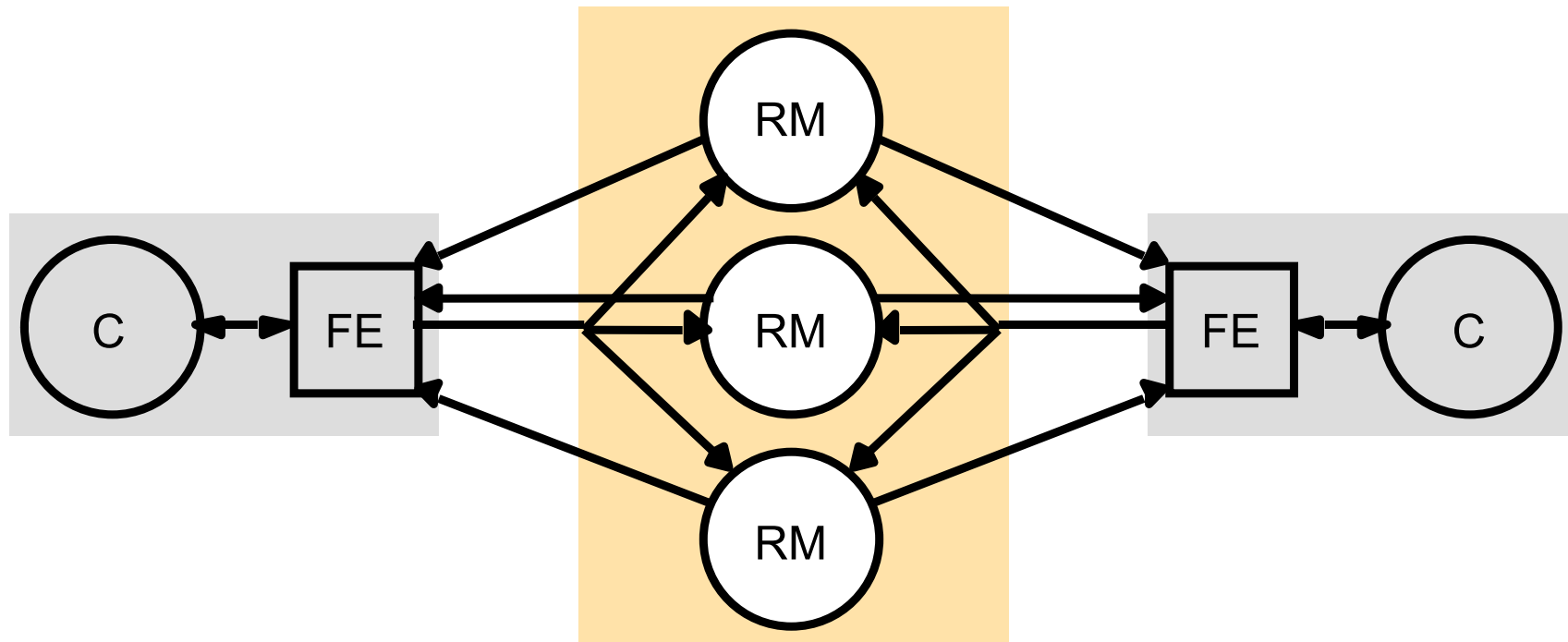


replicação ativa e passiva

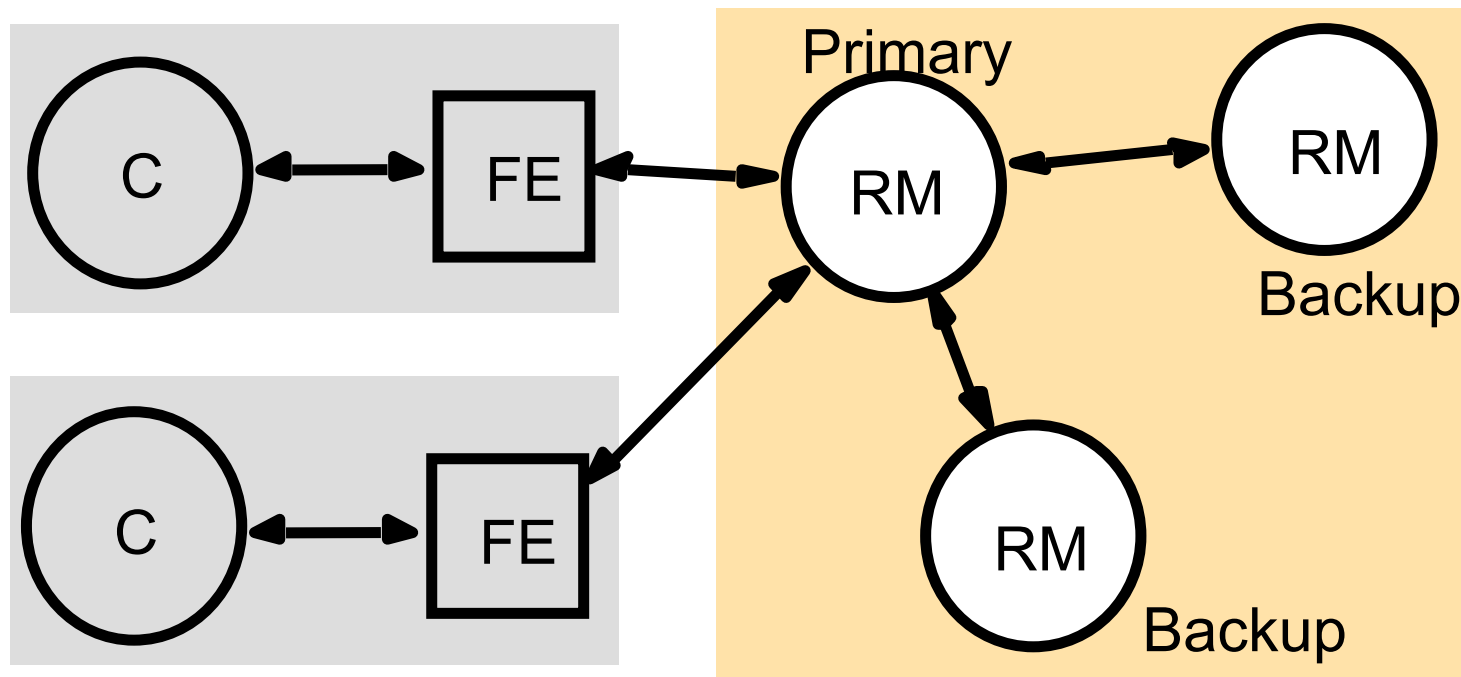
- ativa: gerentes de réplica são máquinas de estado com papéis equivalentes
- passiva: a cada momento uma única cópia primária e uma ou mais cópias secundárias



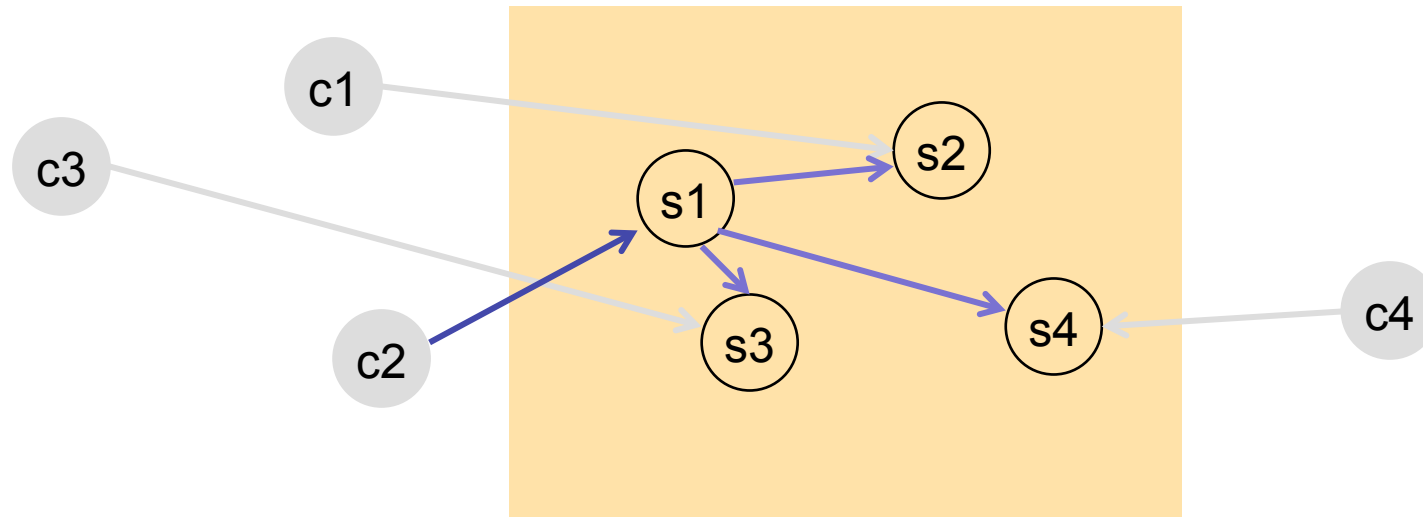
replicação ativa



replicação passiva



abordagens híbridas



- consultas em qualquer cópia
- atualizações apenas na cópia primária Backup

