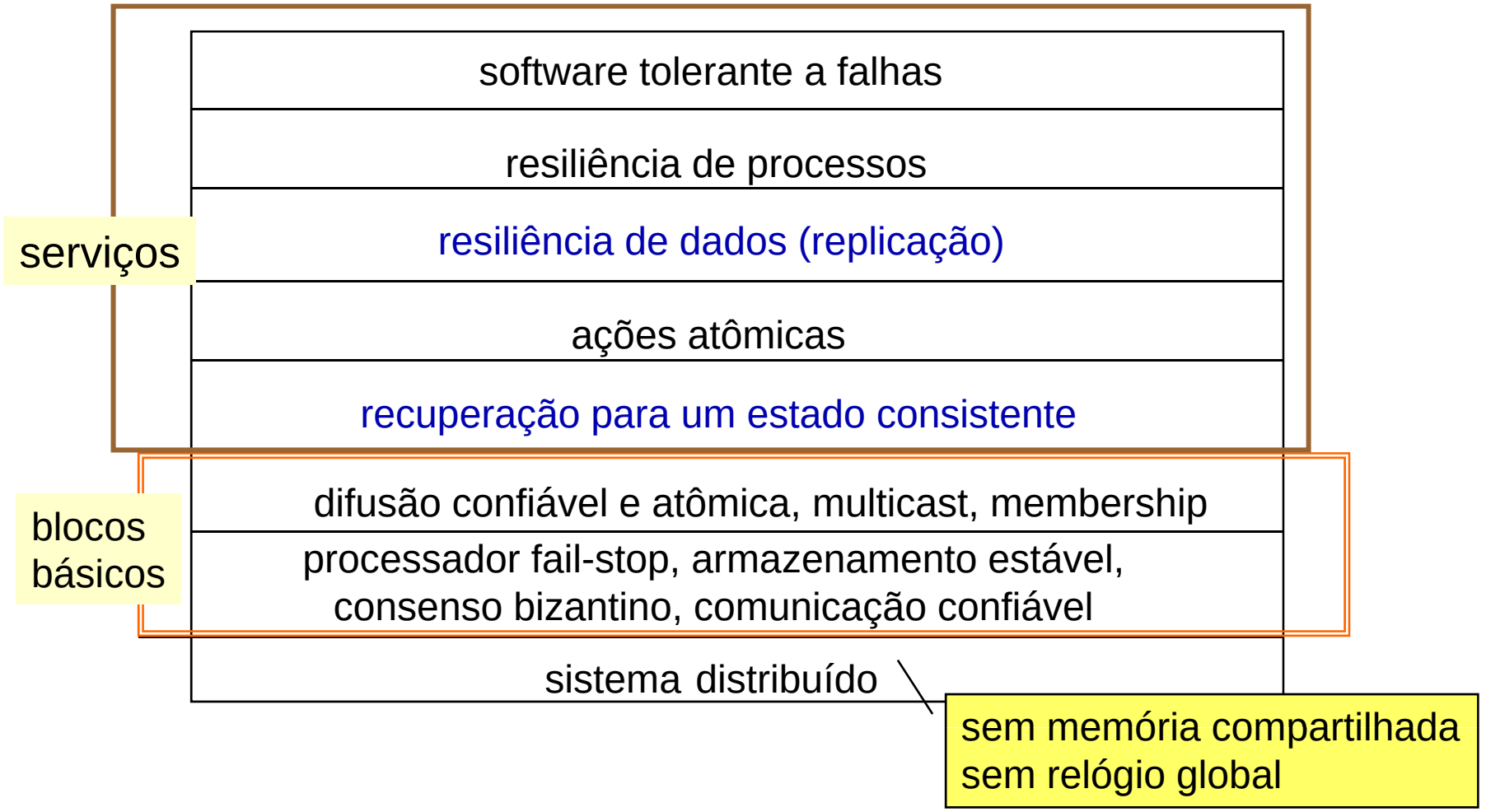
TF em sistemas distribuídos: serviços

UFRGS
Taisy Silva Weber

Níveis - [Jalote 94]



Recuperação

- ✓ restaurar para um estado consistente

  estado global do sistema inclui estados de diferentes processos executando em nodos diversos

- ✓ problemas:
  - ✓ sistema distribuído

    sem relógio global, sem memória comum
  - ✓ estado consistente em sistema distribuído
- ✓ sistemas distribuídos convencionais
  - ✓ usual recuperação por retorno
  - ✓ checkpoint ou ponto de recuperação (PR)
    - ✓ contém **toda** a informação de **todos** os processos executando no nodo

Recuperação por retorno

- checkpointing assíncrono
  - eficiente no avanço
  - demorado no rollback
- checkpointing coordenado
  - checkpointing coordenado em todos os nodos
  - o conjunto dos PRs representa um estado consistente para o sistema
- checkpointing induzido por comunicação
  - info de controle de carona nas mensagens normais
- rollback-recovery
  - com log de eventos não determinísticos

não coordenado nos diferentes nodos

ELNOZAHY, E. N. ; et alli. **A Survey of Rollback-Recovery Protocols in Message-Passing Systems.** ACM Computing Surveys, Sept. 2002, pp. 375–408.

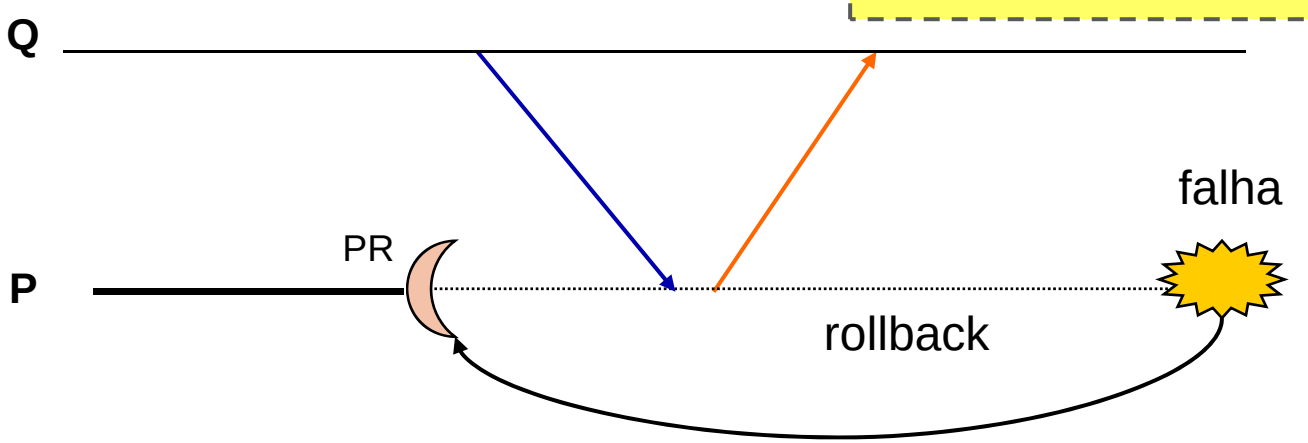
Rollback

sem problemas em um processo isolado

mas em um SD processos trocam mensagens

msg perdida: **receptor** retornou para um ponto anterior ao recebimento da msg

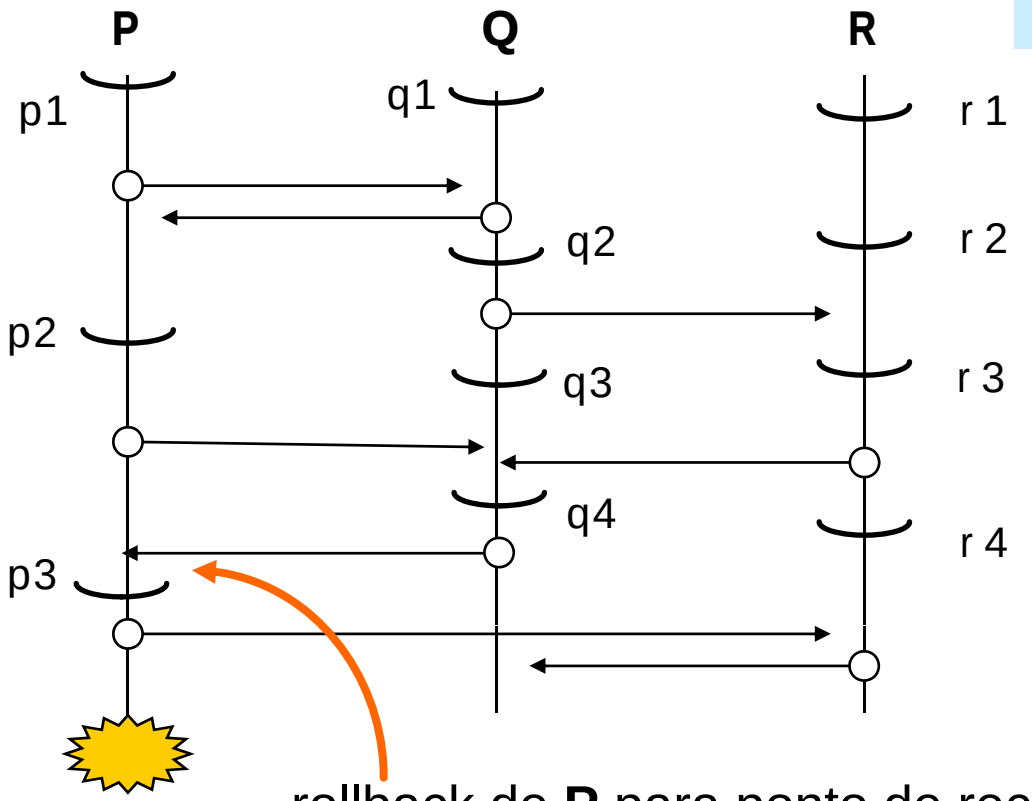
msg orfã: **transmissor** retornou para um ponto anterior ao envio



- ✓ mensagens perdidas: msg enviadas e não recebidas
- ✓ mensagens órfãs: msg recebidas que não foram enviadas

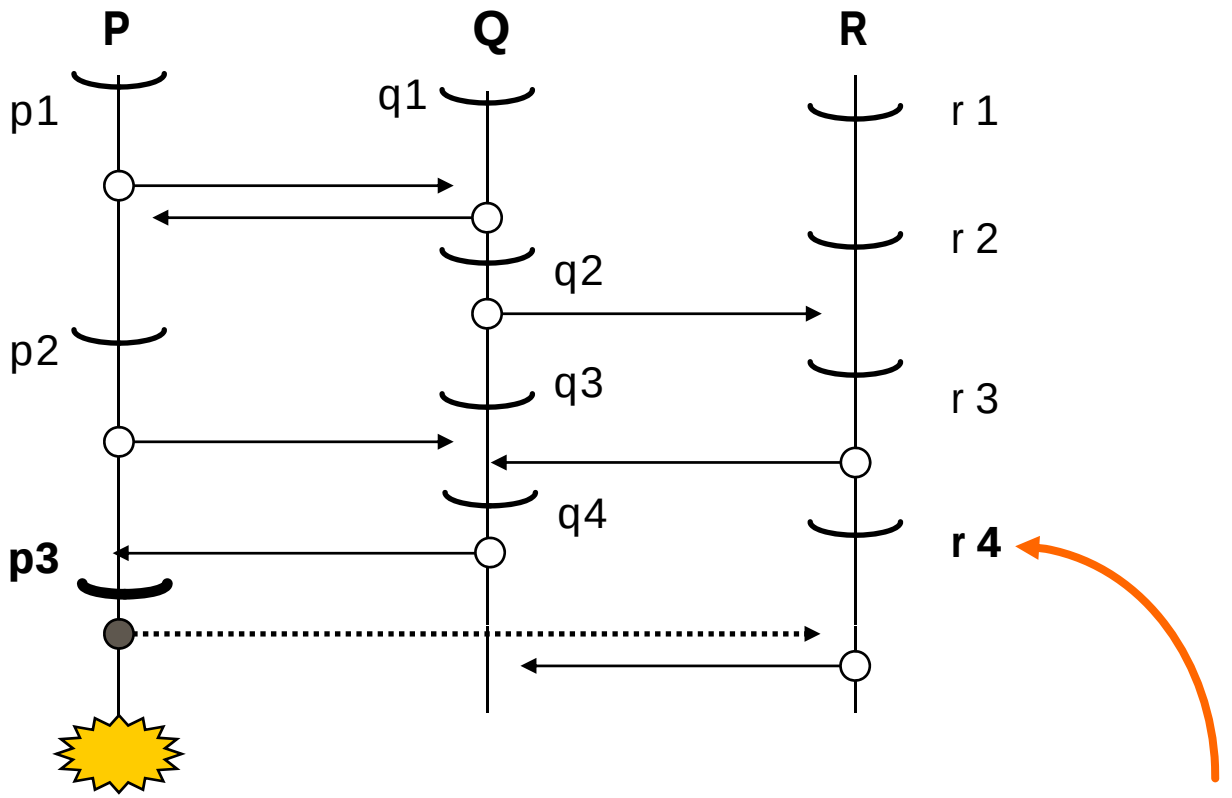
Linha de recuperação

conjunto de *ckpoints* com apenas **um** *checkpoint* por processo, **sem** msgs **órfãs**, **sem** msgs **perdidas**

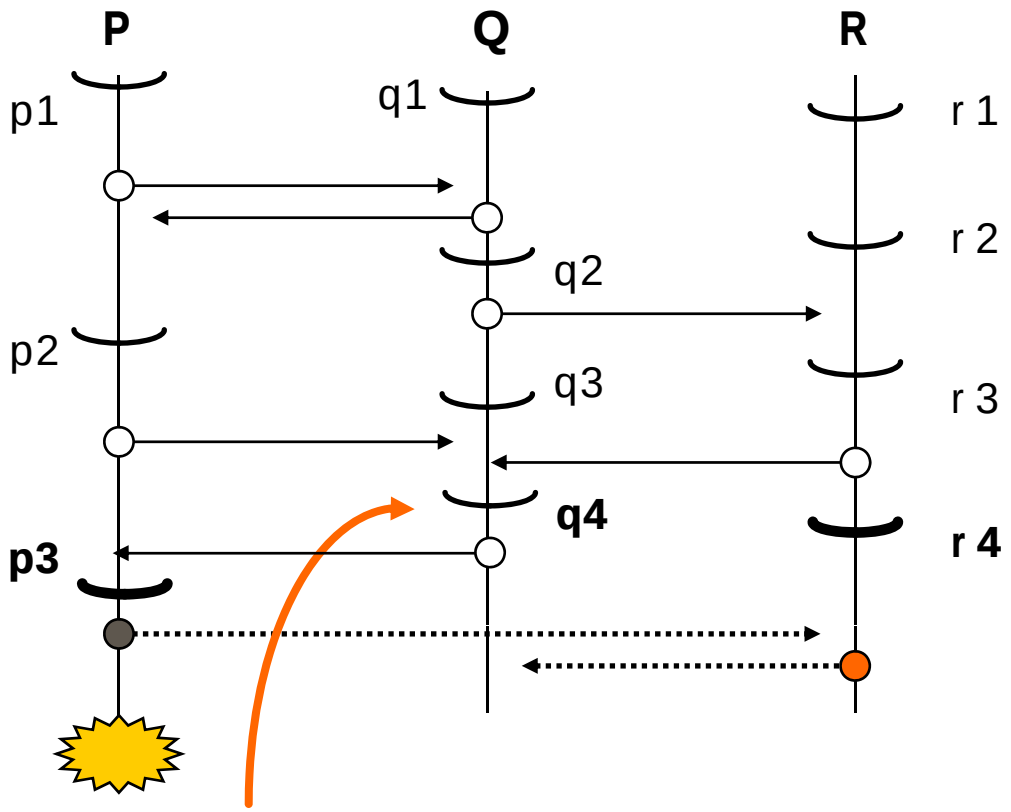


rollback de **P** para ponto de recuperação **p3**

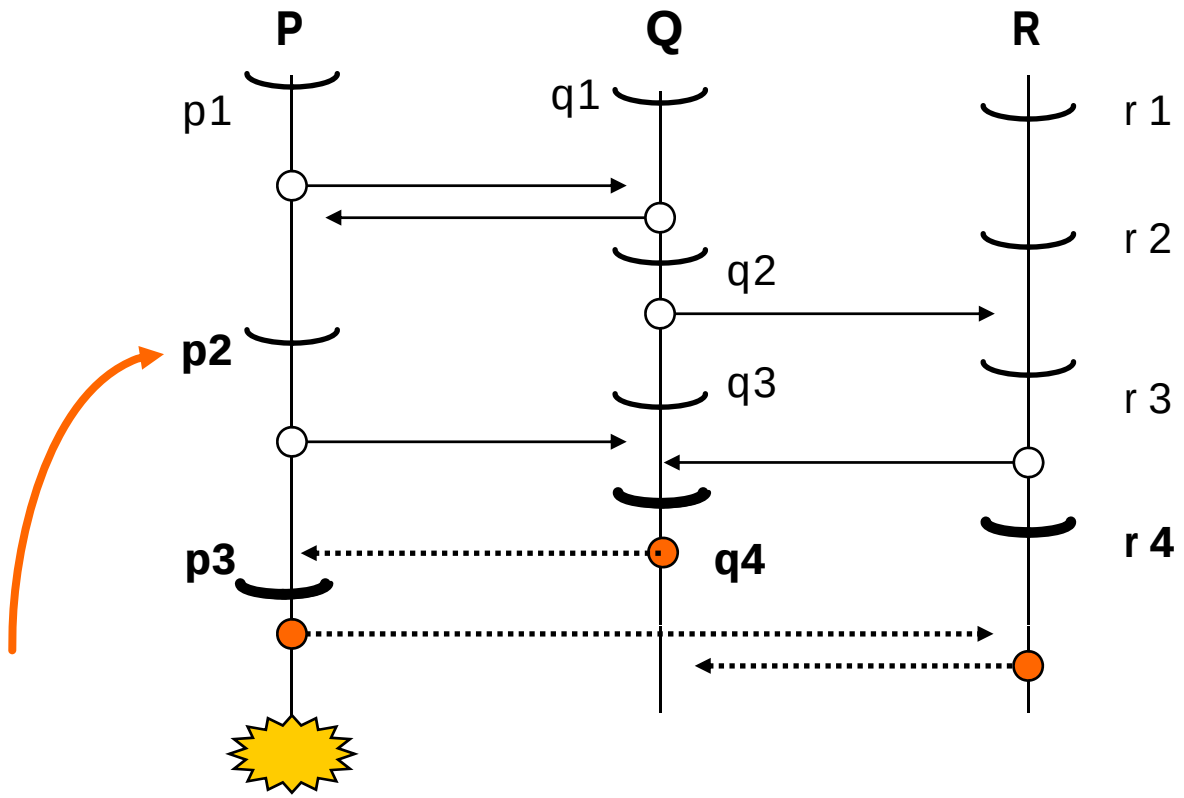
Linha de recuperação



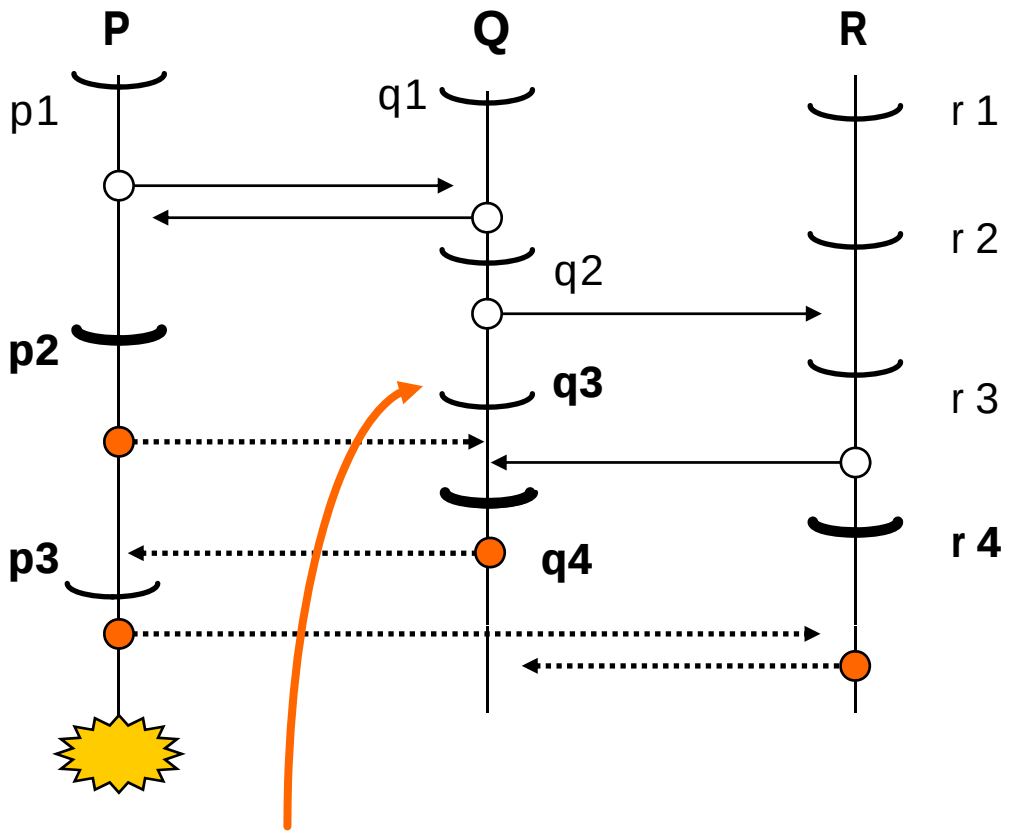
Linha de recuperação



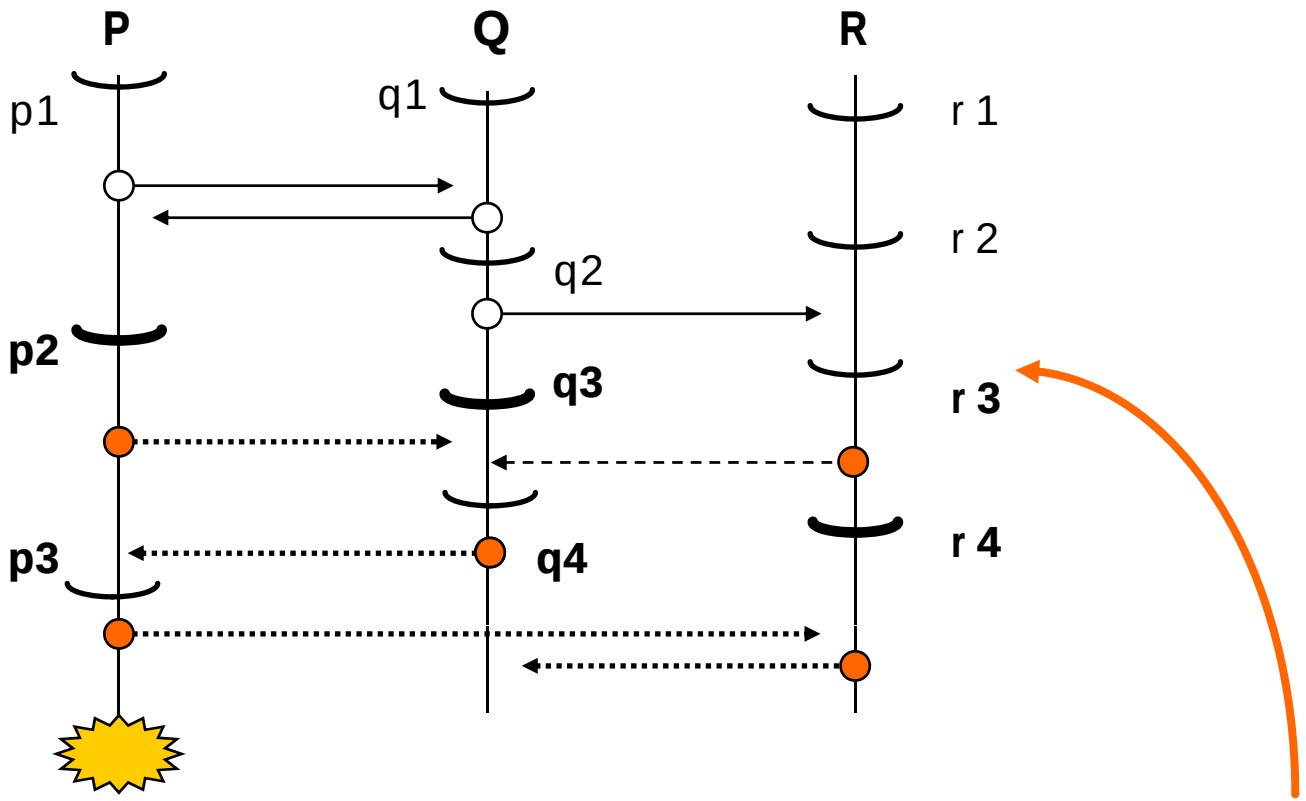
Linha de recuperação



Linha de recuperação

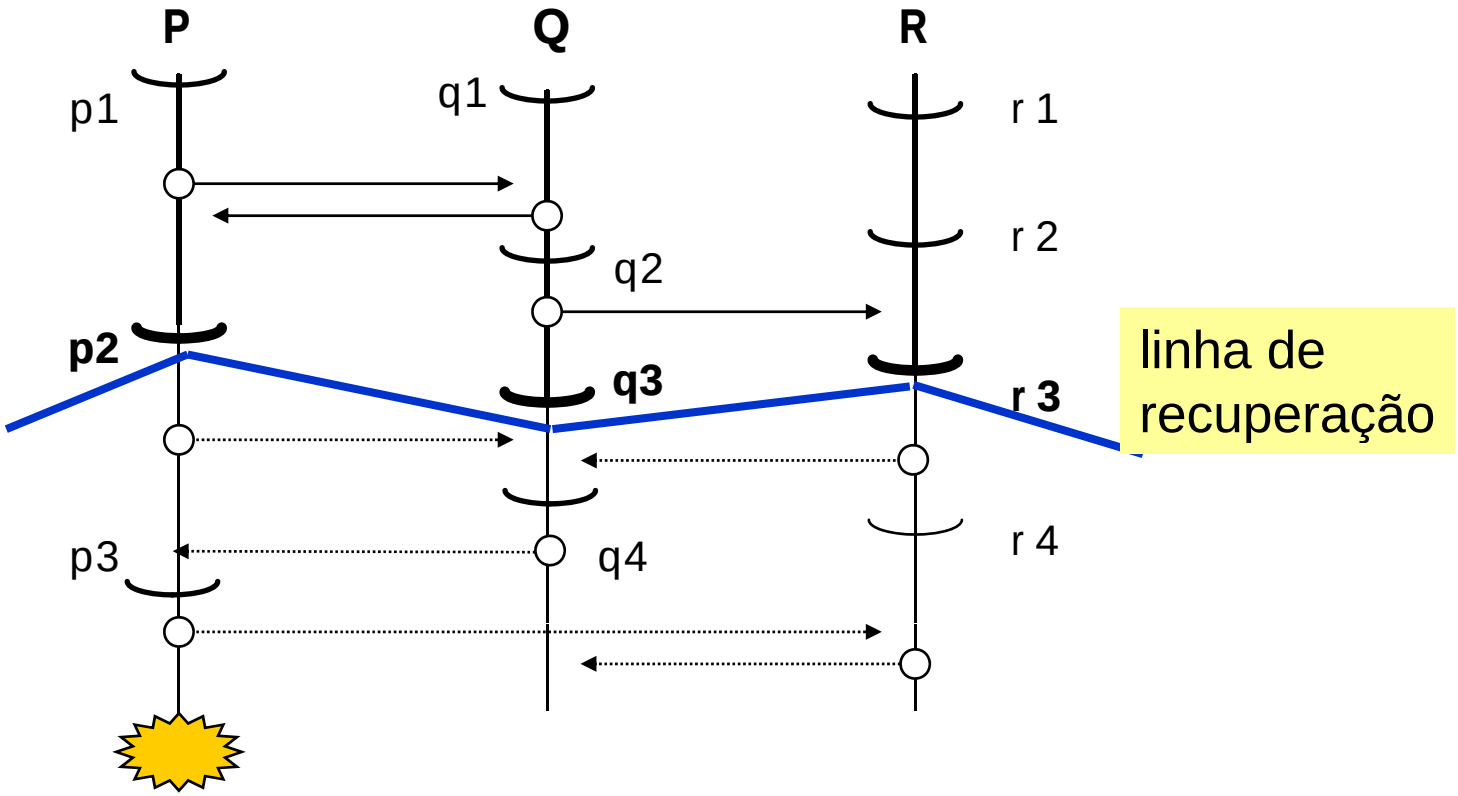


Linha de recuperação



Linha de recuperação

apenas um PR por processo, sem órfãs, sem perdidas



Efeito dominó

avalanche de rollbacks que podem ocorrer durante a recuperação

- perigo real
  - sistemas com estabelecimento de pontos de recuperação independentes e sem restrições a troca de mensagens
  - pode provocar volta a estado inicial
  - típico em ckp não coordenado
- evitando efeito dominó
  - coordenação de checkpointing
  - restrição a comunicação

Recuperação baseada em logs

- premissa
  - todos os eventos não determinísticos podem ser identificados e *logados*
    - eventos não determinísticos podem ser modelados como recepção de mensagem
    - envio de mensagem é um evento determinístico
  - recuperação
    - usa ckps e logs para voltar precisamente pelo mesmo caminho ao estado anterior à falha
      - computação não é perdida
        - mas deve se ter cuidado com respostas (msgs) que alteram o mundo exterior

Replicação de dados

- dados replicados em vários nodos
  - a queda de um ou mais nodos não impede **acesso** aos dados
- novos problemas
  - consistência

cópias diferentes de um objeto devem ser mutuamente consistentes entre si

  - serialização

critério de correção

    - execução concorrente nas réplicas deve ser equivalente a execução correta nos dados lógicos
    - (como se fosse **cópia única**)
  - replicação deve ser transparente ao usuário

Tipos de falhas

- ✓ falhas nos nodos
  - ✓ cópias no nodo ficam inacessíveis
  - ✓ demais cópias são acessíveis e deve ser garantido o acesso e o critério de serializabilidade
- ✓ particionamento da rede
  - ✓ particionamento é difícil de tratar
  - ✓ geralmente são implementadas algumas soluções parciais para casos particulares

modelo físico

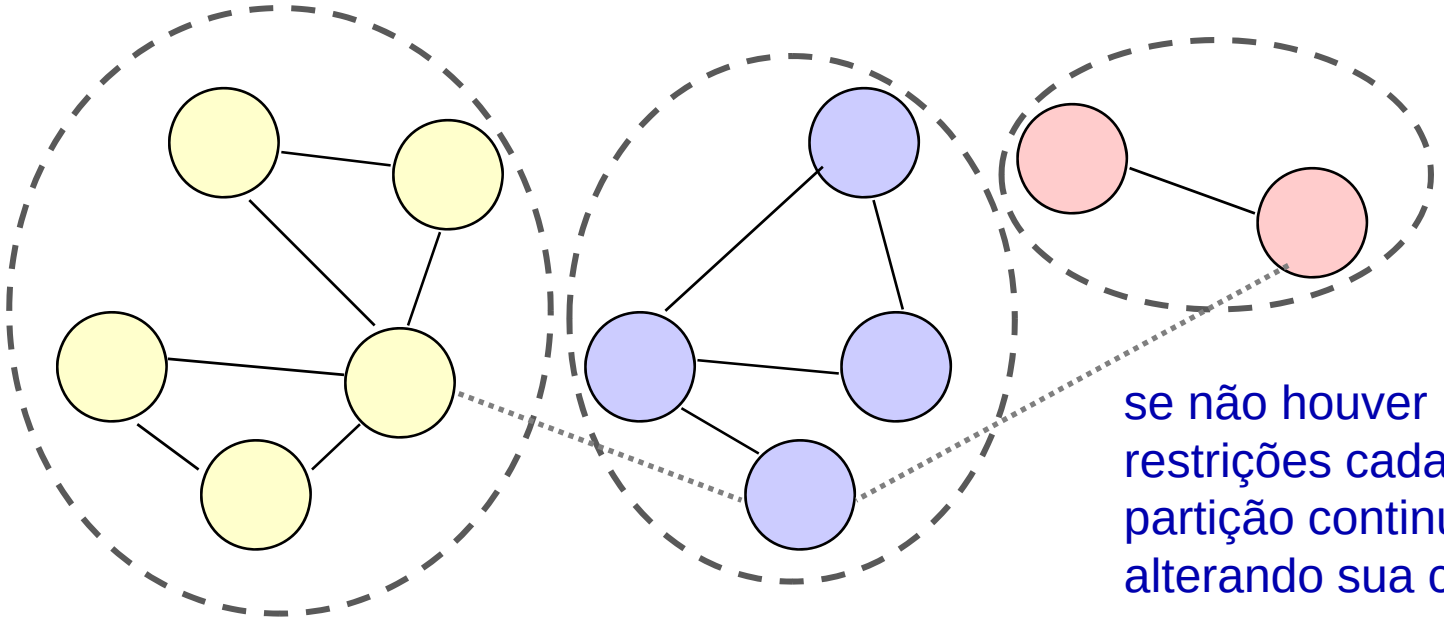
exemplo: cada partição inibe alterações na sua réplica caso não contenha a maioria dos nodos

Particionamento

modelo físico

partição com maior número de nodos

3 partições isoladas que continuam recebendo requisições de escrita dos clientes



se não houver restrições cada partição continua alterando sua cópia

o maior problema é garantir a serializabilidade sem comunicação entre as partições

Estratégias

- protocolo de controle de réplicas
  - otimista
    - sem restrição
      - réplicas podem divergir e usuários podem ver inconsistência
  - pessimista
    - garantia de *consistência forte*
      - réplicas nunca divergem
    - tipos de abordagem pessimista
      - cópia primária
      - réplicas ativas
      - votação

esperança de que operações em partições diferentes não vão conflitar

Abordagem otimista

- estratégia antiga, mas interesse crescente
  - sistema móveis e Internet

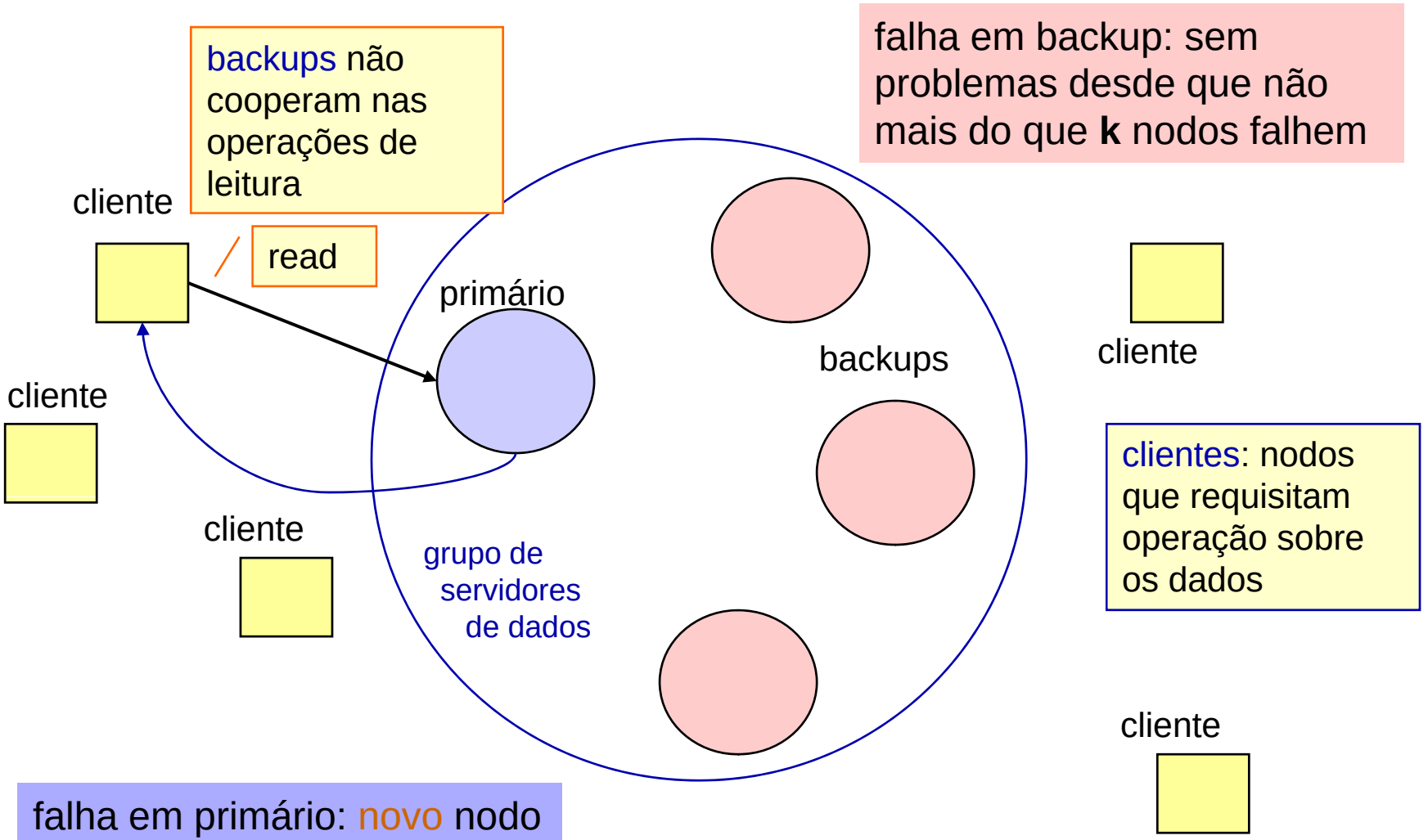
exemplos: DNS, CVS

- vantagens
  - aplicação para sistemas de larga escala
    - pode usar comunicação epidêmica quando a topologia é desconhecida
    - mantém disponibilidade (as custas de inconsistências eventuais)
    - requer pouca sincronização entre réplicas
    - permite nodos operaram autonomamente
      - sem necessidade de estar sempre conectado

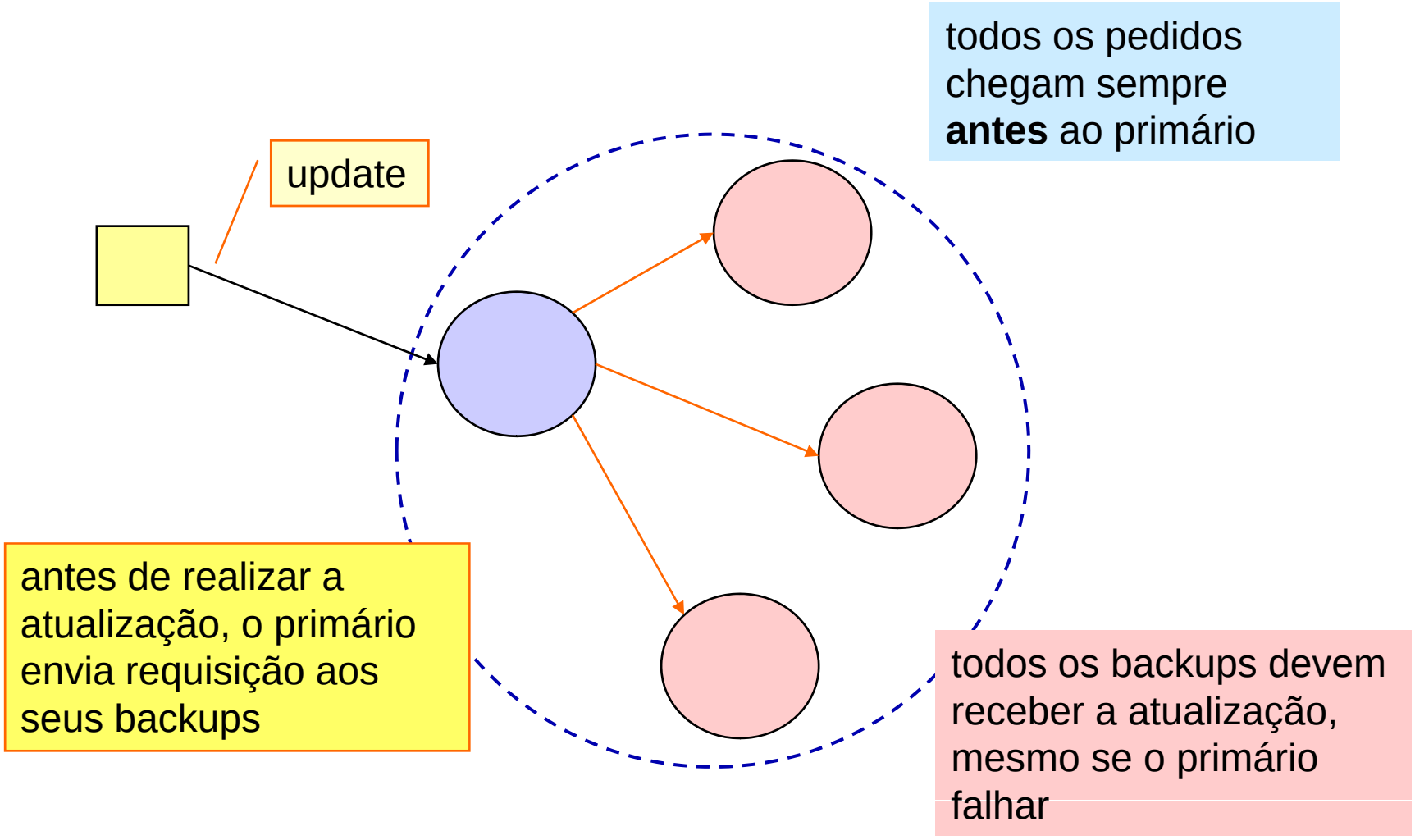
Desafios

- ✓ redes locais
  - ✓ uso popular de primário backup
  - ✓ escala pequena
- ✓ sistemas intensivos quanto a dados
  - ✓ sem necessidade de garantias fortes
    - ✓ redes de entrega de conteúdos
      - ✓ só um nodo altera dados, outros mantém cópias,
    - ✓ P2P (média taxa de atualização)
    - ✓ Data Grids (raras atualizações)
  - ✓ com garantias fortes (ACID)
    - ✓ bancos de dados distribuídos (não escala)

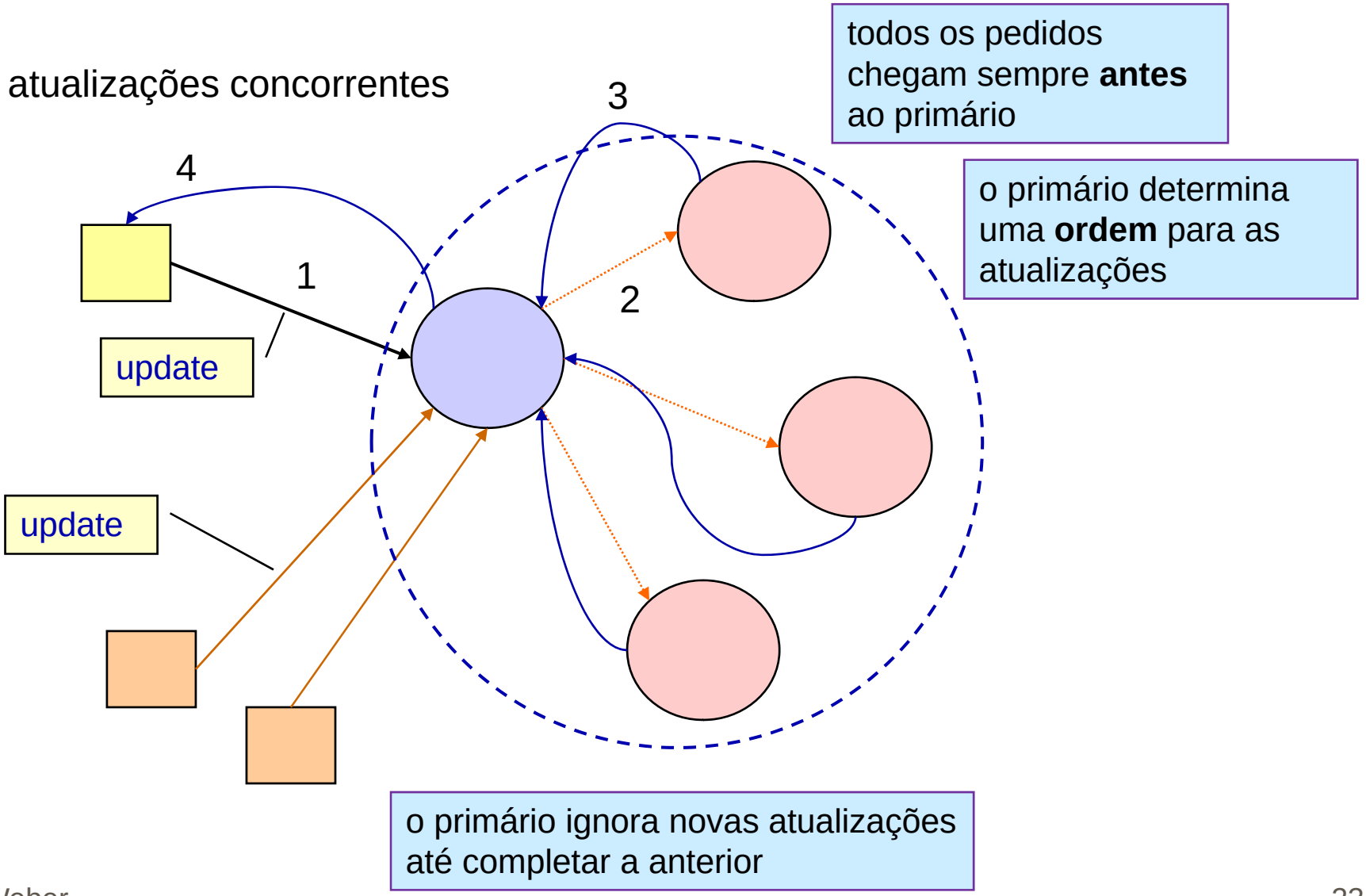
Cópia primária: abordagem pessimista



Atualizações

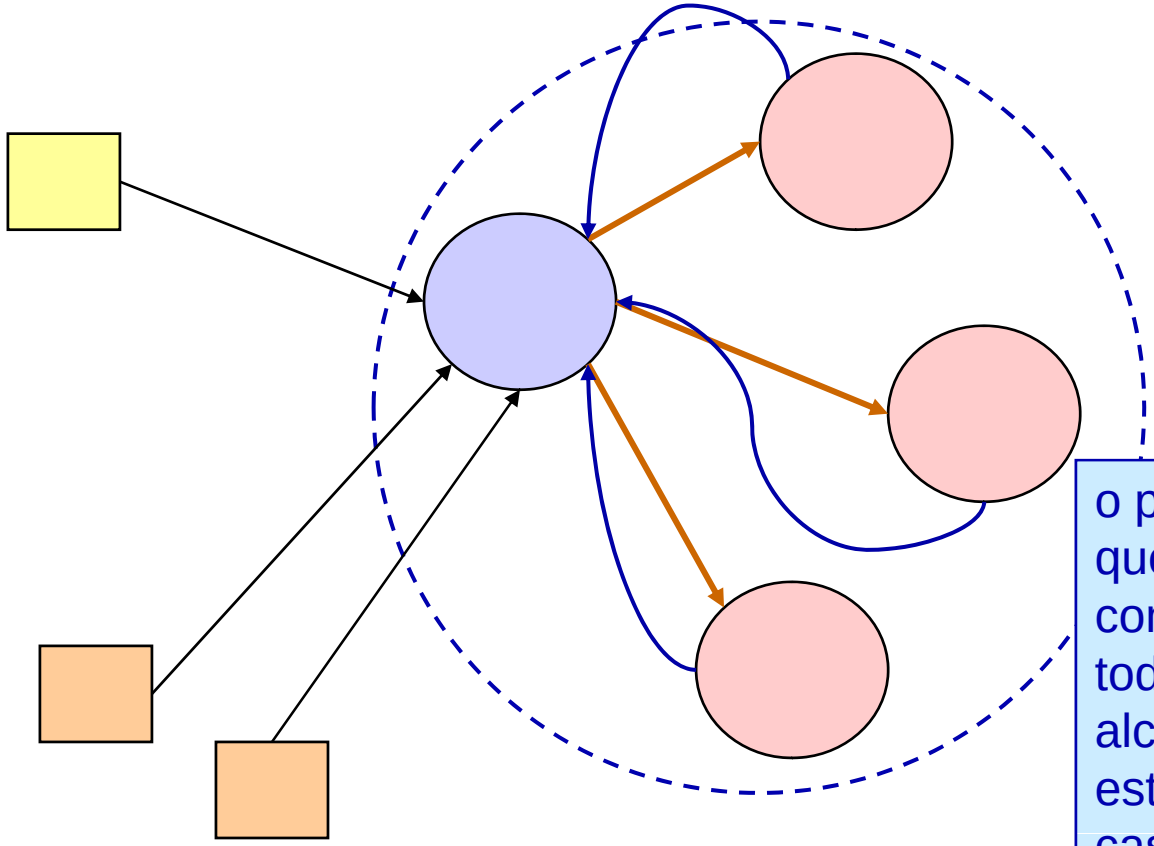


Ordenação de atualizações



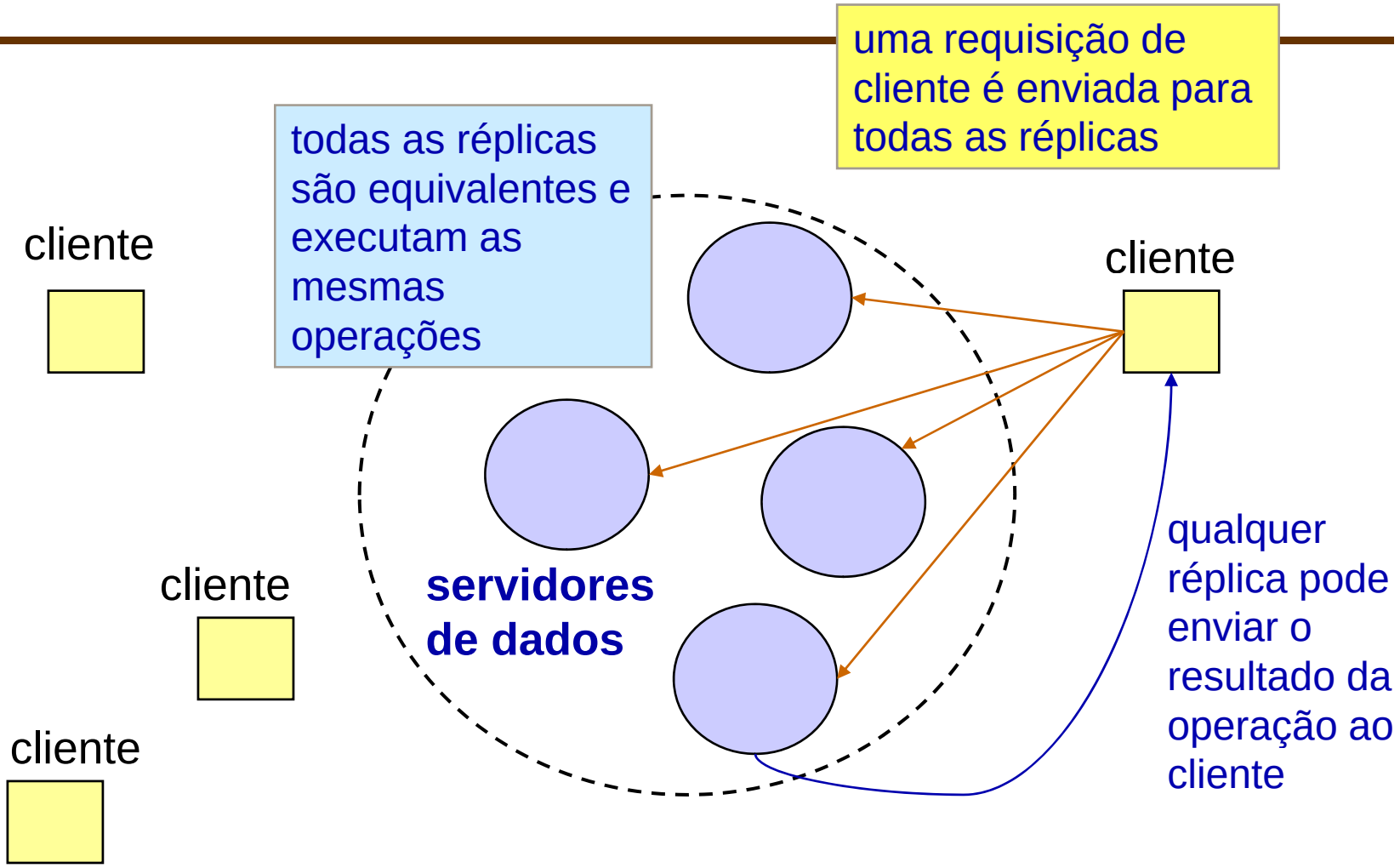
Confiabilidade

a comunicação entre primário e backups pode ser suportada por um **protocolo de multicast confiável**



o protocolo vai garantir que, uma vez que a comunicação iniciou, todos os backups vão alcançar o mesmo estado, mesmo em caso de queda do primário

Réplicas ativas



é essencial que as réplicas sirvam as requisições **no mesmo estado** para preservar a serializabilidade

Serializabilidade com réplicas ativas

- assumido
  - se as réplicas estão no mesmo estado e recebem as requisições na **mesma ordem,** então vão produzir os mesmos resultados
  - devem ser satisfeitas propriedades de:
    - consenso
    - ordem

todas as réplicas operacionais devem receber todas as requisições

todas as réplicas operacionais executam as requisições na mesma ordem

multicast atômico garante consenso (confiabilidade) e ordem

Bibliografia

- JALOTE, P. **Fault tolerance in distributed systems.** Prentice Hall, Englewood Cliffs, New Jersey, 1994

- ELNOZAHY, E. N. ; et alli. **A Survey of Rollback-Recovery Protocols in Message-Passing Systems.** ACM Computing Surveys, Vol. 34, No. 3, September 2002, pp. 375–408.

- YASUSHI SAITO, MARC SHAPIRO. **Optimistic Replication**. ACM Computing Surveys, Vol. 37, No. 1, March 2005, pp. 42–81.