



UFOP – Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas

Lista 3 de Sistemas Distribuídos

Icaro Bicalho Quintão EC-14.1.8083

Questão 1) A Exclusão Mútua Distribuída consiste em garantir o acesso exclusivo de um processo, dentre vários concorrentes a uma seção. Já a Exclusão Mútua é uma técnica usada em programação concorrente para evitar que dois processos ou threads tenham acesso simultaneamente a um recurso compartilhado, acesso esse denominado por seção crítica. A diferença entre elas é que não há o compartilhamento de memória.

Questão 2) Um algoritmo considerado distribuído seus processos executam de forma em que todas suas instancias sejam afetadas da mesma forma, gerando a necessidade deles serem executados em ordem, visto que os processos tem que ser sincronizados.

Questão 3) Não é possível criar um processo detector de falhas em uma rede não confiável, pois as premissas destes processos asseguram que não falham, como por exemplo a comunicação é confiável.

Questão 4) A eleição é necessária para escolher qual dos processos desempenhara um papel de coordenador, sendo que esse processo ao se retirar, deverá ser feita outra eleição para seu substituto, e além disso, para que o eleito seja escolhido é necessário que todos os processos aceitem a escolha.

Questão 5) a) O Algoritmo do Anel funciona da seguinte maneira: Quando um processo nota que não há coordenador, o processo faz eleição para sucessor. Se o ID do remetente é maior do que o ID do destinatário, a mensagem é passada adiante. Cada processo adiciona o seu ID à mensagem de eleição e encaminha-a para o processo seguinte do anel. A eleição termina quando a mensagem chega novamente ao processo inicial. Este envia em seguida uma mensagem coordenador através do anel com informação sobre todos os processos ativos. O coordenador é o processo com maior identificador.

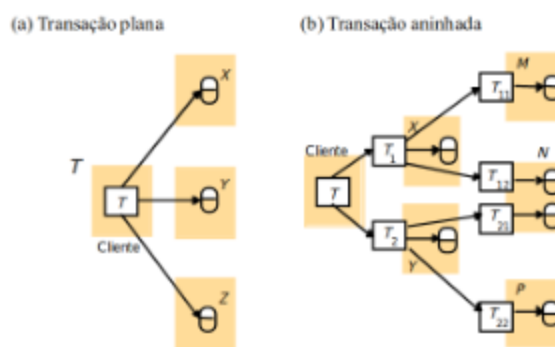
b) O Algoritmo do Valentão trabalha da seguinte forma: se o processo tenta fazer requisição e não consegue, ele inicia a eleição. O processo, então, envia mensagens de eleição para todos

os processos com IDs maiores. Se ninguém responde, o processo vence a eleição e torna-se coordenador. Se algum processo com ID maior responde, ele desiste.

Questão 6) O controle de concorrência é necessário porque ele permite que as transações sejam executadas simultaneamente de forma que os dados se mantem consistentes. Ou seja, evita que se tenha crash ou perda de informações.

Questão 7) No Protocolo de confirmação de duas fases os servidores se comunicam e decidem juntos sobre confirmar ou cancelar uma operação. Elas podem ser planas ou aninhadas.

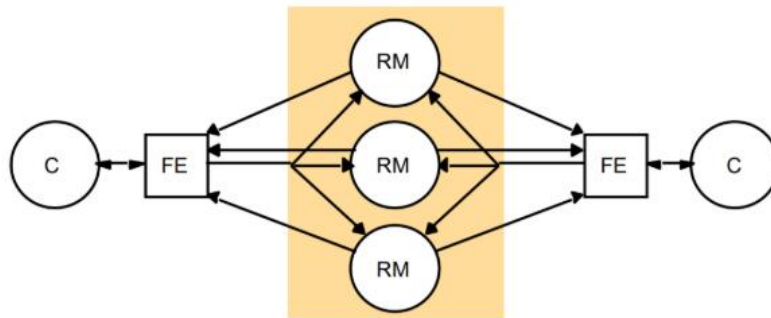
Transações Distribuídas planas e aninhadas



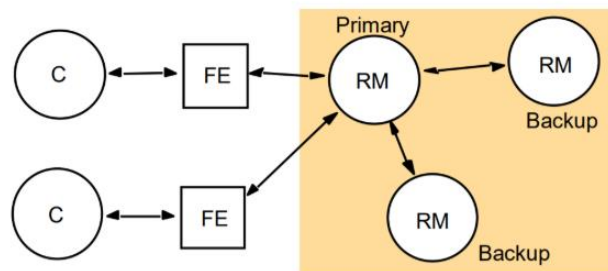
Questão 8) Ocorre quando os processos demoram para votar, sendo assim o coordenador toma uma decisão desconsiderando a opinião destes processos, em sequência tais processos entram em um estado de indecisão.

Questão 9) Usam conhecimento sobre o que em um processo trava o outro, prevendo estado futuro do sistema que possa ocasionar deadlock. O sistema está em um safe-state se não está em deadlock e existe uma ordem de execução dos processos do sistema que garanta que eles possam alocar os recursos necessários para prosseguir e terminar.

Questão 10) Na replicação ativa todos os servidores são atualizados constantemente com as alterações, e podemos assim dizer que temos um processo síncrono. A falha em um servidor será facilmente contornada, já que todos os servidores são atualizados ao mesmo tempo, a requisição do usuário poderá ser respondida por outra máquina. Como na figura abaixo:



Na replicação passiva apenas uma réplica primária recebe, processa e responde a todas as requisições. As outras réplicas são passivas (backup) e assim atualizadas periodicamente através da cópia do servidor central. O problema com a falha do servidor central fará com que um backup assuma o controle das operações, porém, as operações realizadas após o momento da última atualização serão perdidas. Esse modelo de replicação, contudo, requer menor processamento e apresenta melhor desempenho. Como na figura:



Questão 11) A capacidade de linearização consiste na interposição das operações para todos os clientes, para qualquer operação. Um requisito desejável é o requisito de tempo real para que os clientes recebam informações atualizadas, porém esse requisito pode levantar um problema quanto a exequibilidade da capacidade de linearização. É uma medida de correção rigorosa. Já a consistência sequencial diz respeito a interposição das operações, porém não apela para o tempo real. A única noção de ordem relevante é a ordem dos eventos para cada cliente, permitindo assim que a interposição possa embaralhar desde que a ordem de cada cliente não seja violada.

Questão 12) A arquitetura Gossip permite serviço de alta disponibilidade por replicação de dados próximos aos pontos onde os grupos de clientes precisam deles. A GR troca mensagem periodicamente e transmite atualizações recebidas. A vantagem é que o cliente obtém serviço

off-line se tiver GR na partição. E a desvantagem é a escalabilidade, ao aumentar GR, aumenta número de mensagens.

Questão 13) Todas operações devem ser concluídas com êxito, ou não devem ter nenhum efeito na presença de falhas (ou cancelamento).

Questão 14) São elas:

- De acesso: não necessita fornecer a localização dos recursos, ou seja, os programas devem executar os processos de leitura e escrita de arquivos remotos da mesma maneira que operam sobre os arquivos locais, sem qualquer modificação no programa. O usuário não deve perceber se o recurso acessado é local ou remoto.
- De localização: os programas clientes devem ver um espaço de nomes de arquivos uniforme, sem a necessidade de fornecer a localização física dos arquivos para encontrá-los, mesmo que esses arquivos se desloquem entre os servidores.
- De mobilidade: independente dos arquivos se moverem entre servidores, os programas clientes não precisam ser alterados para a nova localidade do grupo de arquivos. Essa característica permite flexibilidade em mover arquivos sem comprometer toda a estrutura, ou ter que refazer links entre programas clientes e o local do arquivo.
- De desempenho: o desempenho da aplicação cliente não poderá ser comprometido enquanto ocorre uma variação dos processos sobre os recursos disponíveis pelos SAD's, isto é, mesmo que haja concorrência no acesso pelos arquivos isso não deve afetar os usuários.
- De escalabilidade: os recursos computacionais podem sofrer alterações para abrigar maior poder computacional ou o ingresso de novos servidores sem prejudicar o serviço.
- Contra falhas: garantir a disponibilidade dos arquivos ininterruptamente e se ocorrerem falhas o programa cliente não deverá saber como elas serão tratadas.
- De replicação: várias cópias dos mesmos arquivos armazenados em locais diferentes para garantir a disponibilidade. A aplicação cliente deverá visualizar apenas uma cópia do mesmo, não necessitando saber a quantidade replicada e o local.

Questão 15) Os arquivos do google tendem a ser muito grandes, pois são atualizados com informações sempre adicionadas ao final do arquivo e não sobrescritas ou substituídas. Ele é

altamente escalável. O Mestre apenas coordena, são os servidores de porção (Chuncks) que fazem todo o trabalho pesado. A arquitetura Mestre/Chuncks pode ser reaplicada, criando super-mestres. Todos os índices ficam em memória.

Questão 16) Dados são replicados para aprimorar confiabilidade ou melhorar desempenho, um dos problemas é manter réplicas consistentes, se uma cópia é atualizada deve assegurar que as outras, em todos os lugares, também serão.