

Perguntas de Teste

1. Explique as vantagens e desvantagens relativas da utilização de formatos de texto (por oposição a formatos binários) para serialização de dados em sistemas distribuídos.

Vantagens: Mais fácil de dar debug; mais fácil de entender; robusto; genérico.

Desvantagens: É necessário fazer parsing; falta de encriptação de dados, ou seja, menos segurança de dados, redundante; tamanho maior.

2. Defina transparência de acesso e explique em que medida é que a invocação remota (RPC) contribui para a obter.

A transparência de acesso ocorre quando não é possível distinguir se o serviço que estamos a usar é local ou remoto

Os sistemas de invocação remota escondem a distribuição de serviço ou seja, permite que o código que temos use um serviço local que possa ser facilmente convertido num código que usa um serviço que reside num sistema remoto.

3. Identifique a principal dificuldade criada pela escala geográfica a aplicações cliente/servidor interativas e explique uma forma de a resolver.

A principal dificuldade é manter uma latência baixa e aceitável para todos os clientes independentemente de onde se encontram geograficamente.

Uma forma de resolver seria com servidores distribuídos (vários servidores implementados em sítios diferentes) e replicação de dados (replicar os dados em vários servidores de maneira a obter uma resposta do servidor mais próximo, ou seja, menos latência).

4. Distinga escala geográfica de escala numérica em sistemas distribuídos e identifique técnicas usadas para as atingir.

Escala numérica -> Dentro de um sistema, mesmo que a quantidade de clientes/pedidos aumente, a performance não diminui. Isto é, se um sistema se diz escalado numericamente, é possível inserir utilizadores e recursos sem a diminuição significativa de performance. Uma técnica para atingir este tipo de escalabilidade é aumentar a quantidade de hardware/servidores para poderem corresponder a uma maior quantidade de utilizadores.

Escala Geográfica -> Dentro de um sistema onde os recursos e os utilizadores se encontrem afastados, a escala geográfica faz com que esta distância não cause delays significativos na comunicação entre ambos. Ou seja, não importa a distância a que a sua localização física fica, a comunicação não é gravemente afetada. Uma das Técnicas para atingir a escala geográfica é a replicação de dados ou a criação de caches para os locais mais próximos dos recursos/utilizadores., assim como os dados se encontram mais próximos, os problemas relativos a latência é diminuída.

5. Explique uma forma de mitigar a incerteza quanto ao tempo de transmissão de mensagens para conseguir sincronizar relógios em sistemas distribuídos.

Uma forma é através do protocolo dos relógios de Lamport. Para implementar este algoritmo, cada processo A mantém um contador $C(i)$. Estes contadores são atualizados conforme as seguintes etapas:

- Antes de executar um evento 'A' incrementa $C(i)$: $C(i) \leftarrow C(i) + 1$;
- Quando o processo A envia uma mensagem M para um processo Pj, define um timestamp $ts(m)$ em M igual a $C(i)$ após ter executado a última ação;
- Na receção de uma mensagem M, o processo Pj ajusta o seu counter local para $C(j) = \max \{C(j), ts(m)\}$, o qual depois executa a primeira etapa e envia a mensagem para a aplicação.

6. Distinga comunicação síncrona de assíncrona em sistemas distribuídos. Dê exemplos de middleware para cada uma delas.

Comunicação Síncrona -> O emissor e o recetor das mensagens estão sincronizados, ou seja, para haver comunicação o cliente e o servidor têm de estar sincronizados. O envio e a receção de mensagens são ambas ações bloqueantes, isto é, quando um cliente/emissor emite uma mensagem este fica bloqueado até obter uma resposta e enquanto o recetor/servidor não recebe uma mensagem este fica também bloqueado.

Comunicação Assíncrona -> Neste caso, o envio não é uma ação bloqueante, ou seja, o emissor pode enviar uma mensagem e continuar a sua execução logo após o envio da mesma, isto porque a mensagem é copiada para um buffer/queue de mensagens. A transmissão destas mensagens ocorre em paralelo com a execução do emissor. No caso da receção de mensagens, a receção tanto pode ser uma ação bloqueante ou não.

Exemplos middleware para síncrona -> Message passing em MOM (middleware orientado a mensagens);

Exemplos middleware para assíncrona -> Message Queuing em MOM.

7. Identifique uma aplicação e descreva sucintamente o funcionamento de um relógio lógico de Lamport num sistema distribuído.

Para sincronizar relógios lógicos, Lamport definiu uma relação Happened-Before(\rightarrow):

- Se A e B são eventos do mesmo processo e A foi executado antes de B, então $A \rightarrow B$.
- Se A é o evento de envio de uma mensagem por um processo e B o evento de recepção dessa mensagem por outro processo, então $A \rightarrow B$.
- Se $A \rightarrow B$ e $B \rightarrow C$, então $A \rightarrow C$.

Para a realização destas relações, é associado uma etiqueta temporal a cada evento do sistema, de forma que se $A \rightarrow B$ então a etiqueta de A é menor que a etiqueta de B, isto é:

- Cada processo tem um relógio lógico associado. O relógio é um contador que é incrementado entre cada dois eventos sucessivos do processo.
- Cada mensagem enviada transporta o instante lógico em que foi enviada.
- Ao receber uma mensagem, o processo ajusta o seu relógio com o instante da mensagem se este último for mais recente.

Um exemplo de aplicação da utilização dos relógios de Lamport é uma Base de Dados replicada em várias cidades.

8. Qual a razão para estruturar uma aplicação distribuída em camadas? Use um exemplo.

Modelo em três camadas (3-Tier), derivado do modelo 'n' camadas, recebe esta denominação quando um sistema cliente-servidor é desenvolvido retirando-se a camada de negócio do lado do cliente. O desenvolvimento é mais demorado no início comparando-se ao modelo em duas camadas porque é necessário dar suporte a uma maior quantidade de plataformas e ambientes diferentes. Em contrapartida, o retorno vem em forma de respostas mais rápidas nas requisições, tanto em sistemas que rodam na Internet ou em intranet, e mais controle no crescimento do sistema.

As três partes de um ambiente modelo três camadas são: camada de apresentação (Interface), camada de negócio e camada de dados. Características esperadas em uma arquitetura cliente-servidor 3 camadas:

- O software executado em cada camada pode ser substituído sem prejuízo para o sistema;
- Atualizações e correções de defeitos podem ser feitas sem prejudicar as demais camadas. Por exemplo: alterações de interface podem ser realizadas sem o comprometimento das informações contidas no banco de dados.

A separação em camadas lógicas torna os sistemas mais flexíveis, permitindo que as partes possam ser alteradas de forma independente. As funcionalidades da camada de negócio podem ser divididas em classes e essas classes podem ser agrupadas em pacotes ou componentes, reduzindo as dependências entre as classes e pacotes; podem ser reutilizadas por diferentes partes do aplicativo e até por aplicativos diferentes. O modelo de 3 camadas tornou-se a arquitetura padrão para sistemas corporativos com base na Web.

9. Explique como funciona um protocolo de exclusão mútua distribuída centralizado. Identifique as suas principais vantagens e desvantagens.

O Protocolo de exclusão mútua distribuída tem base na assunção de que o sistema consiste de n processos e cada processo está no seu processador. Cada processo tem uma zona crítica que requer exclusão mútua. Tem como requisito que se um processo se encontra a executar na zona crítica, então mais nenhum processo se encontra a executar a sua.

O algoritmo centralizado deste protocolo consiste num processo que é escolhido para coordenar o acesso à zona crítica. Qualquer processo que queira executar a sua zona crítica tem de enviar um pedido ao processo coordenador. O coordenador decide se o processo pode entrar na zona crítica e envia uma resposta. Quando o processo recebe a respostas do coordenador, o processo inicia a execução da zona crítica. Quando este acaba a execução da mesma, o processo envia uma mensagem ao coordenador a libertar a zona.

Desvantagens: Se o servidor falhar o sistema cai; num sistema de maiores dimensões um servidor pode se tornar numa performance bottleneck.

Vantagens: Algoritmo justo; os processos executam a zona crítica por ordem de chegada. Garante que apenas um processo entra na zona crítica simultaneamente; não permite starvation, ou seja, nenhum processo espera para sempre; simples e fácil de implementar; requiere apenas 3 mensagens por uso de recurso.

10. Qual a relevância do sistema operativo na resolução do problema de exclusão mútua no modelo de memória partilhada e no modelo de passagem de mensagens?

A função do sistema operativo na resolução do problema de exclusão mútua tem por base uma eficiente gestão dos recursos. Este é responsável por bloquear processos, prevenindo-os de consumir tempo de CPU, enquanto não tiverem permissão para avançar para a região crítica. A ação do sistema operativo neste tipo de problemas no modelo de memória partilhada e no modelo de passagem de mensagens apenas interfere quando os processos são bloqueados e libertados.

Em memória partilhada, tal acontece quando se tenta obter o lock. No caso de passagem de mensagens, os processos são bloqueados desde que enviam os pedidos até à receção de resposta. Em ambos os casos se evitam esperas ativas.

Conceitos extra

Formatos Binários

Vantagens: são mais compactos, logo, são mais eficientes;

Desvantagens: Difícil de dar debug; mais frágeis, isto é, a pequena alteração de um byte do seu conteúdo poderá invalidar toda a mensagem;