Lista 2 - Cap 2 - SD

1. Dê três exemplos específicos e contrastantes dos níveis de heterogeneidade cada vez maiores experimentados nos sistemas distribuídos atuais, conforme definido na Seção 2.2

Os sistemas distribuídos contemporâneos enfrentam crescentes níveis de heterogeneidade, que podem ser observados em três contextos distintos:

```
1. Dispositivos Móveis vs. Dispositivos Fixos: A heterogeneidade surge da
variedade de sistemas operacionais, capacidades de hardware e conectividade
em dispositivos móveis, enquanto dispositivos fixos, como computadores de
mesa, tendem a ser mais homogêneos.

2. Ambientes Urbanos vs. Rurais: Ambientes urbanos apresentam uma ampla
gama de redes de alta velocidade e densidade de dispositivos, enquanto
áreas rurais têm infraestrutura de rede limitada e acesso intermitente à
Internet, introduzindo desafios diferentes.

3. Plataformas de Nuvem e Dispositivos Edge: Heterogeneidade ocorre na
escolha de provedores de nuvem e serviços específicos oferecidos por eles,
além da diversidade de dispositivos Edge, como sensores IoT, que variam em
termos de capacidades e sistemas operacionais.
```

Essa heterogeneidade representa um desafio importante na concepção e operação de sistemas distribuídos contemporâneos.

---

2. Quais problemas você antevê no acoplamento direto entre entidades que se comunicam, que está implícito nas estratégias de invocação remota? Consequentemente, quais vantagens você prevê a partir de um nível de desacoplamento, conforme o oferecido pelo não acoplamento espacial e temporal? Nota: talvez você queira rever sua resposta depois de ler os Capítulos 5 e 6.

   Problemas do Acoplamento Direto (Invocação Remota):

   1. Dependência de Interfaces Fixas.
   2. Fragilidade nas mudanças.
   3. Possível impacto negativo no desempenho.
   4. Limitação na escalabilidade.

   Vantagens do Desacoplamento Espacial e Temporal:

   1. Flexibilidade e evolução independente.
   2. Modularidade e facilidade de manutenção.
   3. Melhor tolerância a falhas.
   4. Facilita a interoperabilidade entre sistemas.

   O acoplamento direto entre entidades que se comunicam, como na invocação remota, pode gerar problemas de dependência e fragilidade, enquanto o desacoplamento espacial e temporal oferece

flexibilidade, modularidade e tolerância a falhas, tornando o sistema mais robusto e adaptável. A escolha entre essas abordagens depende das necessidades específicas do sistema distribuído.

---

3. Descreva e ilustre a arquitetura cliente-servidor de um ou mais aplicativos de Internet importantes (por exemplo, Web, correio eletrônico ou news).

   Na arquitetura cliente-servidor:

   - Na Web, os navegadores são clientes que enviam solicitações HTTP para servidores web, que fornecem conteúdo da web.
   - No correio eletrônico, clientes de e-mail se comunicam com servidores de e-mail para enviar, receber e armazenar mensagens.
   - Em serviços de notícias, como newsgroups, os clientes acessam servidores de notícias para ler e postar mensagens em fóruns de discussão.

---

4. Para os aplicativos discutidos no Exercício 2.1, quais estratégias de posicionamento são empregadas na implementação dos serviços associados?

   Na arquitetura peer-to-peer, todos os processos têm funções semelhantes, promovendo a cooperação entre pares, sem distinção de clientes e servidores. Isso permite escalabilidade ao distribuir tarefas e recursos entre muitos computadores. Estratégias de posicionamento, como mapeamento de serviços em vários servidores, uso de cache, código móvel e agentes móveis, são cruciais para otimizar o desempenho e a confiabilidade de sistemas distribuídos.

   Exemplos contrastantes dos níveis crescentes de heterogeneidade em sistemas distribuídos incluem:

   1. **Sistemas Distribuídos Primitivos**: No início, sistemas eram simples e homogêneos, com poucos nós e serviços limitados.

   2. **Sistemas Distribuídos Adaptados para a Internet**: Com o crescimento da Internet, sistemas tornaram-se globais e heterogêneos, requerendo padrões abertos e middleware.

   3. **Sistemas Distribuídos Contemporâneos**: Atualmente, com a computação móvel, ubíqua e em nuvem, a heterogeneidade aumentou significativamente, envolvendo dispositivos variados em diferentes escalas e ambientes.

   Essas etapas mostram como a heterogeneidade evoluiu ao longo do tempo em sistemas distribuídos.

---

5. Um mecanismo de busca é um servidor Web que responde aos pedidos do cliente para pesquisar em seus índices armazenados e (concomitantemente) executa várias tarefas de Web crawling para construir e atualizar esses índices. Quais são os requisitos de sincronização entre essas atividades concomitantes?

Em um mecanismo de busca que também realiza Web crawling, há requisitos mínimos de sincronização entre as atividades de responder a consultas de usuários e executar o Web crawling. Embora essas tarefas geralmente possam ser executadas de forma independente e simultânea, pode ser necessária alguma sincronização em pontos específicos, como evitar a duplicação de URLs ou o acesso simultâneo a recursos

compartilhados. Isso pode ser alcançado usando técnicas de sincronização, como exclusão mútua ou semáforos, para garantir que as atividades não entrem em conflito.

---

6. Frequentemente, os computadores usados nos sistemas peer-to-peer são computadores desktop dos escritórios ou das casas dos usuários. Quais são as implicações disso na disponibilidade e na segurança dos objetos de dados compartilhados que eles contêm e até que ponto qualquer vulnerabilidade pode ser superada por meio da replicação?

O uso de computadores desktop em sistemas peer-to-peer tem implicações na disponibilidade e na segurança dos objetos de dados compartilhados. A disponibilidade depende da disponibilidade dos computadores dos usuários, tornando-a menos confiável em comparação com servidores dedicados. A segurança pode ser uma preocupação, uma vez que os computadores pessoais podem não ter a mesma segurança robusta. A replicação de dados em vários computadores ajuda a aumentar a disponibilidade e a tolerância a falhas, mas também requer considerações adicionais de segurança devido à multiplicação dos pontos de acesso aos dados. Em resumo, o uso de computadores desktop em sistemas peer-to-peer oferece escalabilidade, mas requer cuidados extras com disponibilidade e segurança.

---

7. Liste os tipos de recurso local vulneráveis a um ataque de um programa não confiável, cujo download é feito de um site remoto e que é executado em um computador local.

Recursos locais vulneráveis a ataques de programas não confiáveis baixados de um site remoto incluem o sistema de arquivos local, a memória do computador, as configurações do sistema, as interfaces de rede, a cache local e outros recursos, dependendo das permissões concedidas ao programa. Medidas de segurança, como assinaturas digitais e verificações de integridade, são essenciais para mitigar esses riscos de segurança.

---

8. Dê exemplos de aplicações em que o uso de código móvel seja vantajoso

O uso de código móvel é vantajoso em diversas aplicações. Ele permite atualizações eficientes de software, distribuição de processamento em dispositivos de usuários, monitoramento de redes, funcionamento offline, otimização de serviços em nuvem, atualizações de dispositivos IoT e aprimoramentos em jogos online. Essa abordagem oferece flexibilidade e eficiência, permitindo que aplicativos se adaptem às necessidades em tempo real e economizando largura de banda em atualizações.

---

9. Considere uma empresa de aluguel de carros hipotética e esboce uma solução de três camadas físicas para seu serviço distribuído de aluguel de carros. Use sua resposta para ilustrar vantagens e desvantagens de uma solução de três camadas físicas, considerando problemas como desempenho, mudança de escala, tratamento de falhas e manutenção do software com o passar do tempo.

Uma solução de três camadas físicas para um serviço de aluguel de carros em uma empresa hipotética envolveria a divisão funcional em camadas de apresentação, lógica de aplicação e dados. Isso proporcionaria uma manutenção mais fácil do software e permitiria escalabilidade, mas também poderia aumentar a complexidade de gerenciamento e introduzir latência adicional devido ao tráfego de rede entre as camadas.

---

10. Dê um exemplo concreto do dilema apresentado pelo princípio fim-a-fim de Saltzer, no contexto do fornecimento de suporte de middleware para aplicativos distribuídos (talvez você queira enfocar um aspecto do fornecimento de sistemas distribuídos confiáveis, por exemplo, relacionado à tolerância a falhas ou à segurança).

O dilema apresentado pelo princípio fim-a-fim de Saltzer no contexto do suporte de middleware para aplicativos distribuídos envolve a questão de onde implementar a tolerância a falhas em um sistema. Por exemplo, ao transferir grandes mensagens de correio eletrônico em uma rede potencialmente não confiável, o protocolo TCP oferece alguma correção de erros, mas não é adequado para interrupções graves na rede. Portanto, o serviço de correio eletrônico adiciona um mecanismo de tolerância a falhas, mantendo um registro do progresso e retomando a transmissão em uma nova conexão TCP quando necessário. Isso ilustra o desafio de decidir em qual nível (middleware ou aplicação) implementar a tolerância a falhas e como equilibrar a confiabilidade em diferentes camadas do sistema distribuído.

---

11. Considere um servidor simples que executa pedidos do cliente sem acessar outros servidores. Explique por que geralmente não é possível estabelecer um limite para o tempo gasto por tal servidor para responder ao pedido de um cliente. O que precisaria ser feito para tornar o servidor capaz de executar pedidos dentro de um tempo limitado? Essa é uma opção prática?

Um servidor simples que responde a pedidos de clientes não pode geralmente estabelecer um limite de tempo para responder aos pedidos, pois os pedidos dos clientes variam em complexidade e tempo de processamento. Para impor um limite de tempo, seria necessário implementar mecanismos de controle, mas isso pode ser impraticável e prejudicar a funcionalidade do sistema. Portanto, é comum permitir que o servidor execute os pedidos sem limitações de tempo e lidar com possíveis atrasos de forma adaptativa.

---

12. Para cada um dos fatores que contribuem para o tempo gasto na transmissão de uma mensagem entre dois processos por um canal de comunicação, cite medidas necessárias para estabelecer um limite para sua contribuição no tempo total. Por que essas medidas não são tomadas nos sistemas distribuídos de propósito geral atuais?

Para estabelecer limites para o tempo gasto na transmissão de mensagens em sistemas distribuídos, medidas como otimização de redes, alocação eficiente de recursos, dimensionamento adequado da largura de banda e controle de jitter são necessárias. No entanto, essas medidas não são amplamente adotadas em sistemas distribuídos de propósito geral devido a desafios financeiros, diversidade de ambientes e complexidade na implementação. Elas são mais comuns em sistemas específicos com requisitos rigorosos de desempenho, como sistemas de tempo real ou transmissão multimídia.

---

13. O serviço Network Time Protocol pode ser usado para sincronizar relógios de computador. Explique por que, mesmo com esse serviço, nenhum limite garantido é dado para a diferença entre dois relógios.

O serviço Network Time Protocol (NTP) é utilizado para sincronizar relógios de computador, mas mesmo com esse serviço, não é possível garantir um limite estrito para a diferença entre dois relógios em sistemas distribuídos. Isso ocorre devido ao desvio de base de tempo e às taxas de desvio dos relógios, que podem variar entre os computadores. Mesmo que os relógios sejam inicialmente ajustados com a mesma hora, eles inevitavelmente se desviarão uns dos outros com o tempo, a menos que sejam regularmente

reajustados. Portanto, não é possível fornecer um limite garantido para a diferença entre dois relógios, pois essa diferença pode aumentar gradualmente devido ao desvio dos relógios em execução nos diferentes sistemas distribuídos.

---

14. Considere dois serviços de comunicação para uso em sistemas distribuídos assíncronos. No serviço A, as mensagens podem ser perdidas, duplicadas ou retardadas, e somas de verificação se aplicam apenas aos cabeçalhos. No serviço B, as mensagens podem ser perdidas, retardadas ou entregues rápido demais para o destinatário manipulá-las, mas sempre chegam com o conteúdo correto. Descreva as classes de falha exibidas para cada serviço. Classifique suas falhas de acordo com seu efeito sobre as propriedades de validade e integridade. O serviço B pode ser descrito como um serviço de comunicação confiável?

Existem dois serviços de comunicação para sistemas distribuídos assíncronos. O serviço A permite mensagens perdidas, duplicadas e retardadas, com verificações de integridade aplicadas apenas aos cabeçalhos. Por outro lado, o serviço B também permite mensagens perdidas e retardadas, mas garante a integridade do conteúdo das mensagens. No entanto, o serviço B pode entregar mensagens tão rapidamente que o destinatário pode não ser capaz de processá-las adequadamente. O serviço A compromete a integridade, enquanto o serviço B compromete a validade das mensagens, tornando-o inadequado para muitas aplicações que exigem confiabilidade na entrega de mensagens. Portanto, o serviço B não pode ser considerado um serviço de comunicação confiável.

---

15. Considere dois processos, X e Y, que utilizam o serviço de comunicação B do Exercício 2.14 para se comunicar entre si. Suponha que X seja um cliente e que Y seja um servidor e que uma invocação consiste em uma mensagem de requisição de X para Y, seguida de Y executando a requisição, seguida de uma mensagem de resposta de Y para X. Descreva as classes de falha que podem ser exibidas por uma invocação.

Ao empregar o serviço de comunicação B, que permite mensagens perdidas, duplicadas, retardadas ou entregues rapidamente, durante uma invocação entre os processos X (cliente) e Y (servidor), várias classes de falhas podem ocorrer. Mensagens podem ser perdidas, levando a perda de requisições ou respostas; mensagens podem ser duplicadas, criando inconsistências nos processos de X e Y; mensagens podem ser entregues com atraso, causando lentidão na comunicação; e, em alguns casos, as mensagens podem ser entregues rapidamente demais, sobrecarregando o processo receptor. Essas falhas podem afetar a comunicação e o desempenho do sistema distribuído, exigindo estratégias de tratamento e recuperação de falhas adequadas para garantir a confiabilidade e integridade das operações.

---

16. Suponha que uma leitura de disco possa, às vezes, ler valores diferentes dos gravados. Cite os tipos de falha exibidos por uma leitura de disco. Sugira como essa falha pode ser mascarada para produzir uma forma de falha benigna diferente. Agora, sugira como se faz para mascarar a falha benigna.

Uma leitura de disco pode apresentar falhas por omissão (falha silenciosa) ou falhas arbitrárias (fornecendo dados incorretos). Para mascarar essas falhas, pode-se usar técnicas de verificação de redundância, como somas de verificação ou códigos de correção de erros, para detectar e, em alguns casos, corrigir erros nos dados lidos do disco. Isso garante que os dados sejam confiáveis e precisos, mesmo em face de falhas na leitura de disco.

---

17. Defina a propriedade de integridade da comunicação confiável e liste todas as possíveis ameaças à integridade de usuários e de componentes do sistema. Quais medidas podem ser tomadas para garantir a propriedade de integridade diante de cada uma dessas fontes de ameaças?

A propriedade de integridade da comunicação confiável visa proteger a integridade das informações em sistemas distribuídos. As principais ameaças incluem falta de reconhecimento confiável da origem das mensagens, ataques aos canais de comunicação e reutilização de mensagens. Para enfrentar essas ameaças, medidas como autenticação, canais de comunicação seguros, criptografia, autenticação e verificação de unicidade de mensagens são recomendadas. Isso garante que as informações permaneçam íntegras e seguras durante a transmissão em sistemas distribuídos.

---

18. Descreva as possíveis ocorrências de cada um dos principais tipos de ameaça à segurança (ameaças aos processos, ameaças aos canais de comunicação, negação de serviço) que poderiam ocorrer na Internet.

Na Internet, diversas ameaças à segurança podem comprometer a integridade e o funcionamento adequado dos sistemas. Essas ameaças incluem a capacidade de invasores enviarem mensagens falsas para processos, ameaçando a autenticidade das comunicações. Além disso, os canais de comunicação podem ser alvos de invasores que tentam copiar, alterar ou injetar mensagens, representando um risco para a privacidade e integridade das informações transmitidas. Por fim, a negação de serviço é uma ameaça em que ataques visam sobrecarregar ou interromper os sistemas ou serviços, tornando-os inacessíveis. Para mitigar essas ameaças, são necessárias medidas como autenticação sólida, uso de canais de comunicação seguros com criptografia, além de firewalls, sistemas de detecção de intrusões e práticas de segurança robustas, como atualizações regulares de software e conscientização sobre segurança.