

Sistemas Distribuídos

Primeira lista de exercícios

Nome: Danilo de Oliveira

Data: 17/04/2018

Capítulo 1

- 1) Compare as definições de Sistema Distribuído feitas por Tanenbaum e por Coulouris (ver slides) e crie uma definição própria que contemple ambas.

Resposta: Sistemas Distribuídos é uma coleção de computadores independentes, interligados através de uma rede de computadores e equipados com um software que permita o compartilhamento de recursos como hardware, software e dados, e apresenta-se aos usuários como um sistema único e coerente.

- 2) Qual o papel do Middleware em Sistemas Distribuídos?

Resposta: O papel do Middleware em Sistemas Distribuídos é fornecer transparência de distribuição de recursos, abstrair a heterogeneidade dos componentes do Sistema Distribuído e oferecer uma visão de sistema único.

- 3) Quais são as metas de um Sistema Distribuído?

Resposta: As metas de um Sistema Distribuído são (i) ocultar distribuição de recursos, (ii) ser aberto e (iii) poder ser expandido.

- 4) Duas motivações para a utilização de um Sistema Distribuído são (i) economia e (ii) colaboração (troca de informações). Cite 3 exemplos de cada uma.

Resposta:

Exemplo 1 de economia: **Sistemas de computação em distribuídos**. Esses sistemas são utilizados para tarefas de computação de alto desempenho e, ao invés de utilizar supercomputadores de custo elevado, utiliza-se um conjunto de computadores para distribuir e paralelizar tarefas de computação.

Exemplo 2 de economia: **Compartilhamento de recursos**. Com os sistemas distribuídos é possível compartilhar um recurso com diversos usuários/clientes. Por exemplo, uma impressora pode ser compartilhada em um sistema distribuído, evitando a compra de uma impressora para cada usuário/cliente.

Exemplo 3 de economia: **Economia de tempo através de replicação**. Uma das técnicas de escalabilidade de um Sistema Distribuído é a replicação de componentes. Com a replicação, podemos conseguir economia no tempo. Por exemplo, um sistema web que possui usuários em diversos países, pode usar a replicação para replicar o servidor e colocar uma réplica em pontos estratégicos para servir aos clientes/usuários com maior rapidez.

Exemplo 1 de colaboração: **Groupware**. O Groupware são sistemas que permitem pessoas dispersas geograficamente a trabalharem de forma colaborativa. Por exemplo, vídeo conferência ou edição de arquivos de forma colaborativa.

Exemplo 2 de colaboração: **BitTorrent**. O BitTorrent é um protocolo de rede que permite fazer download ou upload de arquivos de forma distribuída. Nesse sistema, os pares se conectam através de um servidor chamado "Tracker" e, a partir daí, começam a colaborar.

Exemplo 3 de colaboração: **Google Maps**. O Google Maps possui uma funcionalidade que apresenta aos seus usuários a condição do trânsito. Os usuários que utilizam do serviço, colaboram enviando informações através do seu dispositivo.

5) Descreva brevemente os tipos de **transparência**. Dê exemplos.

Resposta:

- **Transparência de Acesso:** Oculta as diferenças de representação de dados e a forma como os dados são acessados. Por exemplo, o serviço de armazenamento chamado Dropbox. Nesse sistema, os dados podem estar armazenados em diferentes dispositivos com diferentes arquiteturas e representações de dados.
- **Transparência de Localização:** Oculta a localização de um determinado arquivo. Por exemplo, um site na web que possui uma URL www.exemplo.com. O endereço da URL oculta a localização física do servidor que responde a essa URL.
- **Transparência de Migração:** Oculta o fato de que um recurso pode ser movido para uma outra localização. Um exemplo é o servidor de arquivos do Google Drive. Um usuário pode possuir arquivos em seu Google Drive que estão armazenados em determinado dispositivo fornecido pelo Google. Em um determinado momento, os dados do usuário podem ser migrados para outro servidor sem que o usuário perceba.
- **Transparência de Relocação:** Oculta o fato de que um recurso pode ser movido para outra localização enquanto o mesmo está em uso. Um exemplo de relocação, é o uso de dispositivos móveis em uma rede sem fio. Os usuários podem se mover em um local sem se desconectar temporariamente.
- **Transparência de Replicação:** Oculta o fato de que um recurso é replicado. Por exemplo, a Rede de Distribuição de Conteúdo – CDN. As CDNs possuem servidores que armazenam réplicas de conteúdo de outros sites no cache e entrega aos visitantes baseado na localização.
- **Transparência de Concorrência:** Oculta o fato de que um recurso é compartilhado por mais de um usuário e que é acessado concorrentemente. Um exemplo de transparência de concorrência é um servidor da Amazon que fornece serviços de computação para usuários. Esses servidores podem ser utilizados por vários usuários concorrentemente sem que os usuários percebam a concorrência.
- **Transparência de Falha:** Oculta falhas e recuperação de um recurso quando elas ocorrem. Um exemplo é quando um usuário tenta acessar uma página web, onde

ocorre um erro na requisição e uma nova requisição é feita com sucesso, sem que o usuário perceba a recuperação da falha.

6) Qual a diferença entre migração e relocação?

Resposta: A migração ocorre quando um recurso é movido de um lugar para o outro quando o recurso não está em uso. A relocação ocorre quando o recurso é movido de um lugar para o outro enquanto em uso.

7) Qual a transparência mais difícil de ser implementada e porque?

Resposta: A transparência mais difícil de ser implementada é a transparência de falha, pois é muito difícil distinguir quando um recurso está morto ou insuportavelmente lento.

8) Por que nem sempre é uma boa ideia visar à implementação do mais alto grau de transparência possível?

Resposta: Existem ocasiões em que é melhor expor a distribuição em vez de tentar ocultá-la. Um exemplo seria uma aplicação de internet que tenta contatar um servidor repetidas vezes antes de desistir. Nesse caso, tentar mascarar uma falha de servidor, poderia reduzir a velocidade do sistema como um todo, sendo que poderia ser melhor desistir mais cedo ou permitir ao usuário cancelar as tentativas de contatar o servidor.

9) O que é um sistema distribuído **aberto** e quais os benefícios que a abertura proporciona?

Resposta: Um sistema distribuído é aberto quando oferece serviços de acordo com regras padronizadas que descrevem a sintaxe e a semântica desses serviços. Em sistemas distribuídos, os serviços são especificados através de interfaces descritas como uma **linguagem de definição de interface (Interface Definition Language – IDL)**.

Um sistema distribuído aberto proporciona benefícios como facilidade para configurá-lo; completude e neutralidade em suas interfaces, permitindo a possibilidade de interoperabilidade e a portabilidade; e facilidade em adicionar novos componentes ou substituir existentes sem afetar os que continuam no mesmo lugar.

10) Diferencie **Interoperabilidade** e **Portabilidade**.

Resposta: **Interoperabilidade** é a capacidade de dois sistemas ou componentes de fornecedores diferentes coexistirem e trabalharem em conjunto, através de serviços especificados por um padrão comum.

Portabilidade é a capacidade de uma aplicação desenvolvida para um sistema distribuído A poder ser executada, sem modificação, em outro sistema distribuído B diferente que implementa as mesmas interfaces de A.

11) Com relação a **Escalabilidade** de um sistema distribuído:

a. Descreva 3 **dimensões** (medida de escalabilidade);

Resposta: Escalável em relação ao tamanho, em termos geográficos e em termos administrativos.

b. Cite 3 exemplos de **limitações** (problemas de escalabilidade);

Resposta:

- Exemplo 1: Existem casos em que é inevitável o uso de um único servidor. Nesse caso, existe o problema de serviços, dados ou algoritmos centralizados. Sendo assim, todos os usuários/clientes tem de conectar a esse único servidor.
- Exemplo 2: Um exemplo de problema de escalabilidade geográfica é o fato de que redes de longa distância são lentas em comparação com as LANs. Por esse motivo, é uma boa ideia considerar a utilização da comunicação assíncrona ao invés da comunicação síncrona.
- Exemplo 3: Outro exemplo é quando um sistema distribuído precisa ser expandido para outros domínios. Nesse caso, é necessário resolver problemas de políticas conflitantes em relação à utilização de recursos, gerenciamento e segurança.

c. Cite 3 **técnicas** (de escalabilidade).

Resposta: Utilização de comunicação assíncrona, distribuição e replicação.

12) Aponte os desafios (ciladas) que tornam a construção de um Sistema Distribuído mais complexa que a de um sistema convencional.

Resposta: Ao desenvolver um sistema distribuído pela primeira vez, pode-se adotar falsas premissas como: a rede é confiável, a rede é segura, a rede é homogênea, a topologia não muda, a latência é zero, a largura de banda é infinita, o custo de transporte é zero e só há um administrador.

13) Qual a diferença entre a computação Cluster e a Grid?

Resposta: A computação em Cluster normalmente é utilizada para programação paralela tornando-se uma alternativa para supercomputadores e consiste em um conjunto de PCs semelhantes (heterogêneos) conectados por uma rede local de alta velocidade. Cada nó do Cluster executa o mesmo Sistema Operacional.

A computação em Grid possuem alto grau de heterogeneidade e nenhuma premissa pode ser adotada em relação a hardware, SO, redes, etc. Os recursos de diferentes organizações são reunidos para permitir a colaboração de um grupo de pessoas ou instituições e essa colaboração ocorre sob a forma de uma organização virtual.

14) Quais as propriedades de uma transação?

Resposta: Uma transação possui propriedades **ACID** (atômicas, consistentes, isoladas e duráveis).

15) Explique o funcionamento de uma transação aninhada.

Resposta: Uma transação aninhada é composta por diversos níveis de subtransações que são executadas em paralelos em máquinas diferentes proporcionando alto desempenho. Se uma transação de nível baixo não for comprometida, todas as transações devem ser revertidas.

16) O que é um monitor de processamento de transações (monitor TP)?

Resposta: Um monitor de processamento de transações permite que uma aplicação acesse vários servidores de banco de dados oferecendo a ela um modelo de programação transacional. Dessa forma, o Monitor TP serve como um servidor de transações para uma aplicação.

17) Dê exemplos de sistemas pervasivos.

Resposta: Smartphones e dispositivos móveis, redes de sensores e sistemas domésticos.

Capítulo 2

18) No contexto de um sistema distribuído, defina o que são:

a. Componentes de software;

Resposta: Componente de software é uma unidade modular que possui interfaces requeridas e fornecidas bem definidas que são substituíveis dentro de seu ambiente.

b. Estilos arquitetônicos;

Resposta: Estilo arquitetônico é uma arquitetura formulada em termos de componentes, modo de conexão entre os componentes e configuração dos elementos para formar um sistema.

c. Arquitetura de sistemas.

Resposta: Arquitetura de sistemas é a especificação final de uma arquitetura de software, incluindo a especificação de máquinas em que os componentes serão executados.

19) Descreva a organização dos **estilos arquitetônicos** abaixo:

a. Arquitetura em camadas;

Resposta: Essa arquitetura é composta por N camadas hierárquicas dispostas de forma vertical, na qual a camada N envia requisições para a camada N-1, que por sua vez envia uma resposta para a camada N.

b. Arquitetura baseada em objetos;

Resposta: Essa arquitetura é composta por objetos que correspondem aos componentes. Esses componentes são conectados e se comunicam por meio de chamada de procedimento remota ou invocação de método remoto.

c. Arquitetura centrada em dados;

Resposta: Possui processos que se comunicam através de um repositório comum.

d. Arquitetura baseada em eventos.

Resposta: Essa arquitetura é caracterizada por componentes e um barramento de eventos. Nessa arquitetura, um componente é realiza o papel de publicar eventos no barramento. O evento publicado é recebido por componentes que se inscreveram no barramento de eventos. Em sistemas distribuídos, os sistemas que realizam a propagação de eventos são denominados sistemas publicar/subscrever.

A arquitetura baseada em eventos também é caracterizada por seus processos estarem desacoplados ou referencialmente desacoplados.

A arquitetura baseada em eventos pode ser combinada com a arquitetura centrada em dados. Essa combinação é conhecida como espaços compartilhados de dados.

20) Com relação a **arquitetura de sistema**, existem 3 tipos principais (ver abaixo). Descreva a organização de cada um delas:

a. Centralizadas;

Resposta: Possui o modelo cliente-servidor que podem resultar em uma arquitetura de duas divisões físicas ou três divisões físicas. Em duas divisões físicas, há uma distinção entre apenas dois tipos de máquinas, sendo clientes ou servidores. Em três divisões físicas, um servidor pode agir como um cliente (p. ex. Como um cliente de um servidor de banco de dados). Muitas dessas arquiteturas centralizadas visam dar suporte ao acesso de usuários a banco de dados e seguem o modelo cliente-servidor possuindo três níveis, sendo (i) nível de interface de usuário, (ii) nível de processamento e (iii) nível de dados.

b. Descentralizadas;

Resposta: Nas arquiteturas de sistema descentralizadas, existem dois tipos de distribuição: vertical e horizontal. Na distribuição vertical, os componentes logicamente diferentes são distribuídos em máquinas diferentes. Na distribuição horizontal, os clientes e servidores são distribuídos em partes logicamente equivalentes.

c. Híbridas.

Resposta: As arquiteturas híbridas podem combinar soluções cliente-servidor com arquiteturas descentralizadas. Uma classe importante de arquiteturas híbridas é os sistemas de bordas. Arquiteturas híbridas são bastante utilizadas em sistemas distribuídos colaborativos.

21) Observe a Figura 2.5 (a)-(e) do livro texto (Tanenbaum, S.D., 2ª ed.).

a. Dê exemplos de sistemas para as alternativas de organizações de cliente-servidor apresentadas nos itens (a), (c) e (e).

Resposta:

(a) PCs 386 ou 486 com 4 MB de memória e sem disco rígido, conectados a um Windows NT terminal server (servidor), como terminais burros.

(c) Aplicativos de página única (single page application – SPA).

(e) Spotify.

- b. Defina (i) clientes gordos (fat clients) e (ii) clientes magros (thin clients), e indique em que alternativa estaria cada um.

Resposta:

(i) Clientes gordos: São clientes que executam boa parte das tarefas. Quanto mais tarefas o cliente tiver em relação à interface gráfica, aplicação e banco de dados, mais cliente gordo ele é.

(ii) Clientes magros: O clientes magros são opostos aos clientes gordos, ou seja, realizam menos tarefas, fazendo com que boa parte das tarefas sejam realizadas pelo servidor.

- c. Qual das alternativas é mais dependente da plataforma do cliente? E por que?

Resposta: A alternativa que mais é dependente da plataforma do cliente é a alternativa (e), pois a plataforma do cliente possui mais tarefas para serem executadas do que o servidor.

- 22) O que é uma arquitetura cliente-servidor de **três divisões**?

Resposta: Uma arquitetura cliente-servidor de três divisões é uma arquitetura que possui cliente e servidor, e o servidor pode atuar também como um cliente. Por exemplo, um servidor de aplicação que faz requisições a um servidor de banco de dados.

- 23) Qual a diferença entre uma **distribuição vertical** e **distribuição horizontal**?

Resposta: Na distribuição vertical, os componentes logicamente diferentes são distribuídos em máquinas diferentes. Na distribuição horizontal, os clientes e servidores são distribuídos em partes logicamente equivalentes.

- 24) Procure na Web e ilustre com exemplos reais sistemas com as seguintes arquiteturas:

- a. Cliente-Servidor;

Resposta: Facebook.

- b. Peer-to-Peer;

Resposta: BitTorrent.

- c. Proxy Server;

Resposta: Servidor de proxy em uma empresa atuando como um Firewall.

- d. Código Móvel;

Resposta: JavaScript

- e. Cliente Magro (Thin Client);

Resposta: Terminal burro.

- f. Agente Móvel;

Resposta: Sistemas de computação em Grade.

g. N-Camadas ?

Resposta: Sistema web utilizando diversas camadas para interface, aplicação e banco de dados.