



Unidade 02: Modelos de Sistemas

Professor: Raimundo Viégas Junior
rviegas@ufpa.br

Créditos: Prof. Josivaldo Araújo

josivaldo@ufpa.br

Instituto de Ciências Exatas e Naturais
Faculdade de Computação
Bacharelado em Sistemas de Informação



Introdução

Sistemas que são produzidos para trabalhar em ambiente real em qualquer lugar do mundo, precisam ser projetados para funcionar corretamente no maior número possível de circunstâncias e em face das muitas possíveis dificuldades e ameaças tais como:

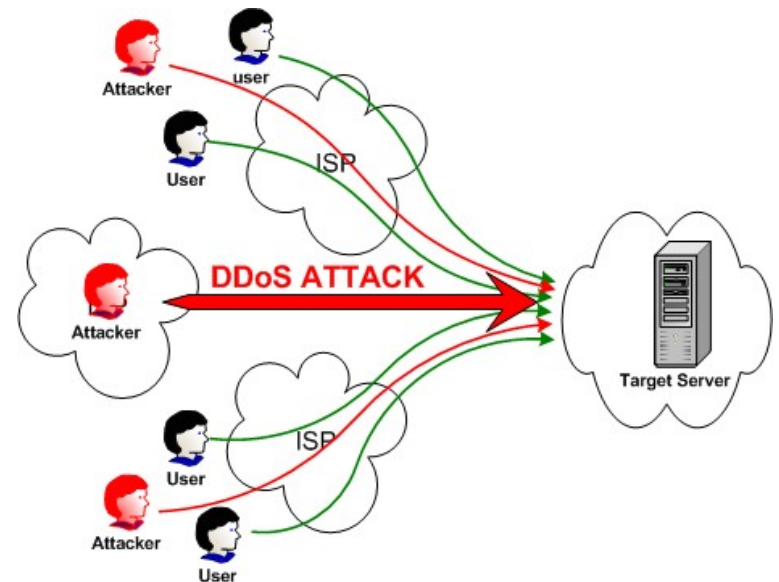
Grande variação de modo de uso: As partes componentes dos sistemas estão sujeitas a grande variação da carga de trabalho.

Grande variação de ambientes: Um sistema distribuído precisa acomodar *hardwares*, sistemas operacionais e redes heterogêneas.

Introdução

Problemas Internos: Relógios não sincronizados, atualização de dados conflitantes, muitas formas de falhas de hardware e software envolvendo componentes individuais do sistema.

Ameaças Externas: Ataque a integridade e segurança dos dados. Ataque do tipo - *denial of service* - excessiva requisição de serviços.





Modelos de Arquitetura

Um modelo de arquitetura, primeiro simplifica e abstrai as funções dos componentes individuais de um sistema distribuído e então considera:

- # A localização dos componentes através da rede de computadores, procurando definir padrões úteis para a distribuição dos dados e da carga de processamento.
- # A relação entre os componentes, isto é, suas regras funcionais e os padrões de comunicação entre eles.

A simplificação inicial é realizada para classificar os processos.



Modelos de Arquitetura

Os processos podem ser:

- » Processos Servidores;
- » Processos Clientes;
- » Processos *Peer*;

Essa classificação dos processos identifica:

- A responsabilidade de cada um;
- Ajuda a taxar a carga de processamento;
- Determinar o impacto de falhas em cada um deles.



Modelos de Arquitetura

Camadas de Software

O termo arquitetura de software referia-se a estruturação do software como camadas ou módulos em um único computador;

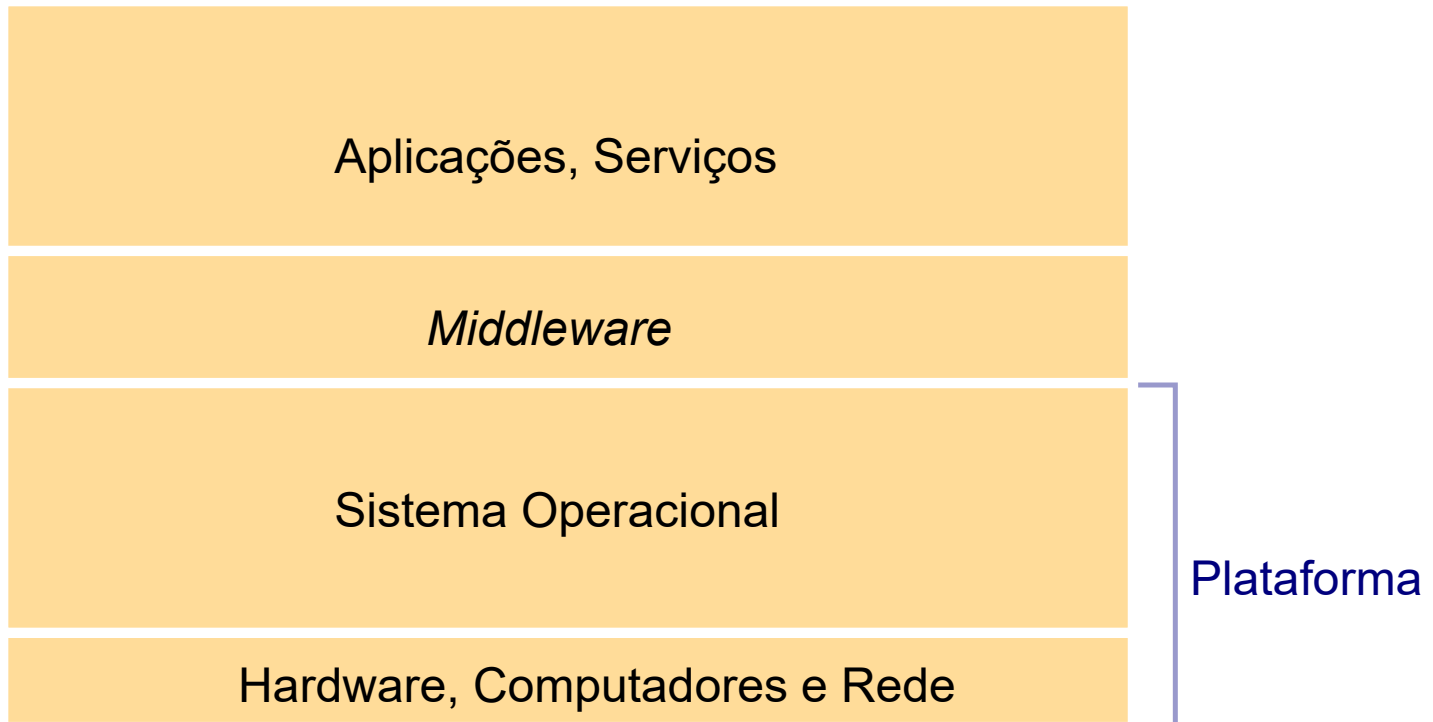
Mais recentemente, em termos de serviços oferecidos e requisitados entre os processos localizados no mesmo ou em diferentes computadores.

- # Um processo servidor é aquele que aceita requisições de outros processos.
- # Um serviço distribuído pode ser provido por um ou vários processos servidores interagindo com cada outros e com o processo cliente, na ordem, para manter a consistência do sistema.

Modelos de Arquitetura

Camadas de Software

Esta visão de orientação à processo e à serviço pode ser expressa em termos de camadas de serviços.





Modelos de Arquitetura

Camadas de Software

Plataforma:

- » Os níveis mais baixos, camadas de hardware e software, são frequentemente referidos como a plataforma para sistemas distribuídos e aplicações.
- » Provêem serviços para as camadas acima, que são implementadas independentemente em cada computador ⇒ Facilitando a comunicação e coordenação entre processos.

Como exemplo: Intel x86/Windows, Sun SPARC/Sun, Intel x86/Solaris, Powerpc/MacOS, Intel x86/Linux.



Modelos de Arquitetura

Camadas de Software

Middleware:

- » É a camada de software cujo propósito é criar uma máscara para a heterogeneidade e prover um modelo de programação conveniente para os programadores de aplicações.
- » É representado por processos e objetos em um conjunto de computadores que interagem entre si para implementar suporte a comunicação e compartilhamento de recursos para aplicações distribuídas.



Modelos de Arquitetura

Camadas de Software

Middleware:

- » Em particular, alcança o nível das atividades de comunicação dos programas de aplicação através de suporte e abstrações, tais como:
 - » Chamada de Métodos Remotos (RMI);
 - » Comunicação entre grupos de processos;
 - » Notificação de Eventos;
 - » Replicação de dados compartilhados;
 - » Transmissão de dados multimídia em tempo real.



Modelos de Arquitetura

Camadas de Software

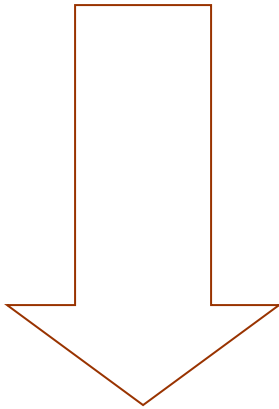
Middleware:

- ✓ Como ambientes mais empregados atualmente, tem-se:
 - » Chamada Remota de Procedimentos Sun RPC;
 - » Sistema de Comunicação em grupo ISIS;
- ✓ Produtos e Padrões orientados a objetos:
 - » CORBA;
 - » Java RMI;
 - » DCOM (padrão Microsoft);

Modelos de Arquitetura

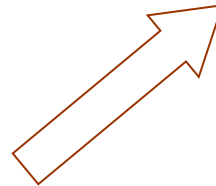
Camadas de Software

Middleware



AMBIENTES E SUPORTES PARA
PROGRAMAÇÃO DISTRIBUÍDA.

RPC's, RPC/DCE, ONC-RPC-SUN
JAVA-RMI, CORBA, COM/DCOM
Transações distribuídas, Monitores
Transacionais





Modelos de Arquitetura

Arquitetura de Sistemas

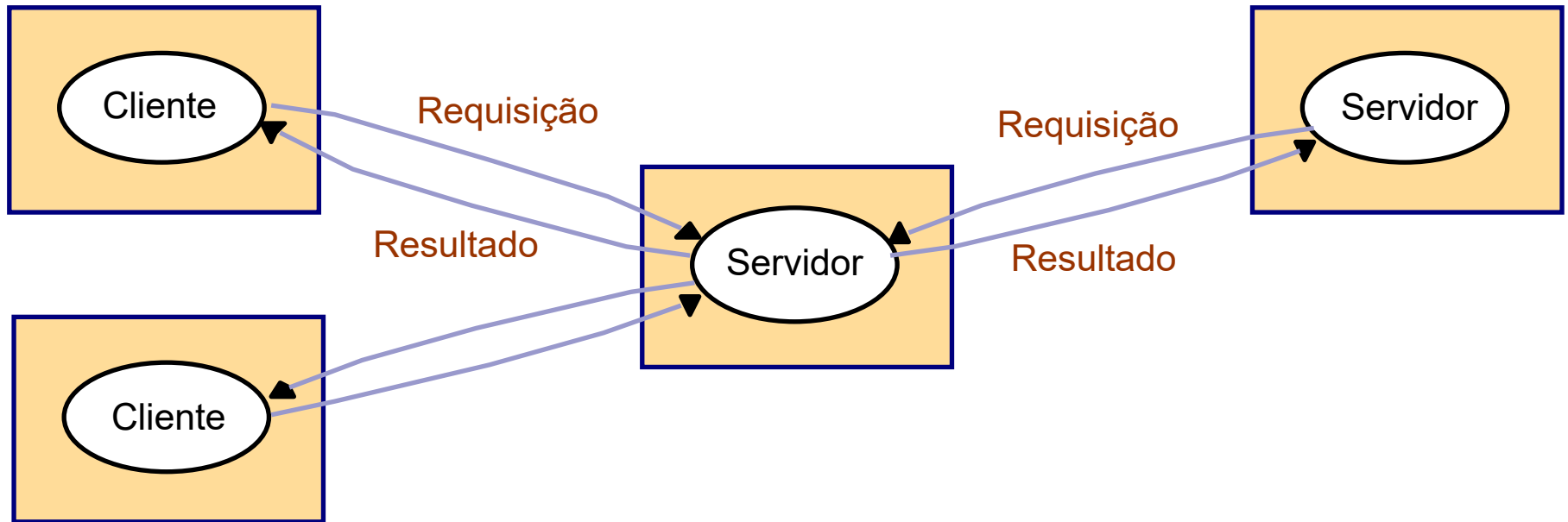
- ✓ A divisão de responsabilidade entre os componentes do sistema (**aplicação, servidor e outros processos**) e a localização dos componentes nos computadores na rede é talvez o mais evidente aspecto do projeto de sistemas distribuídos.
- ✓ Os principais modelos de arquitetura são:
 - » **Cliente-Servidor;**
 - » **Serviços atendidos por Múltiplos Servidores;**
 - » **Servidores Caches e Proxy;**
 - » **Processos *Peer*;**

- » É a arquitetura mais frequentemente citada quando se discute sistemas distribuídos;
- » A estrutura é simples, um processo cliente interage com processos servidores individuais em computadores separados de forma a acessar os recursos compartilhados que o servidor gerência;
- » Um servidor pode, por sua vez, ser cliente de outro servidor.

Modelos de Arquitetura

Arquitetura de Sistemas

Cliente-Servidor



Legenda:

Processos:



Computador:

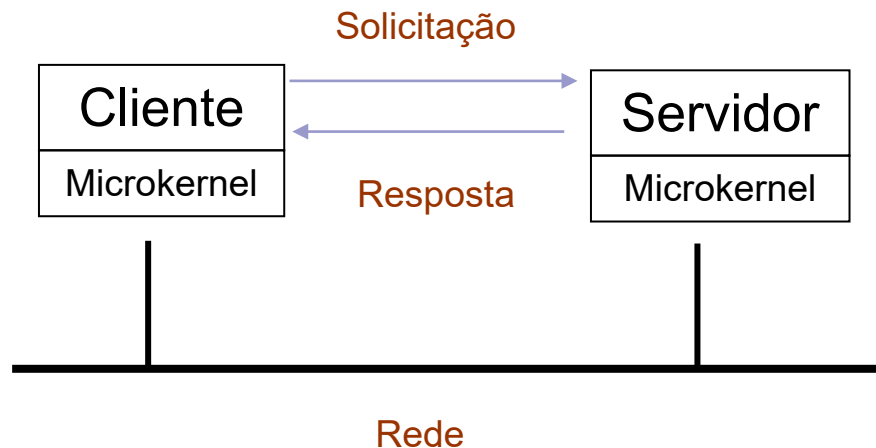


Modelos de Arquitetura

Arquitetura de Sistemas

Cliente-Servidor

- ✓ O modelo cliente/servidor é baseado em um protocolo muito simples, sem conexão, do tipo solicitação/resposta.



- ✓ Nenhum tipo de conexão deve ser estabelecido antes do envio da solicitação, nem desfeito após a obtenção da resposta.

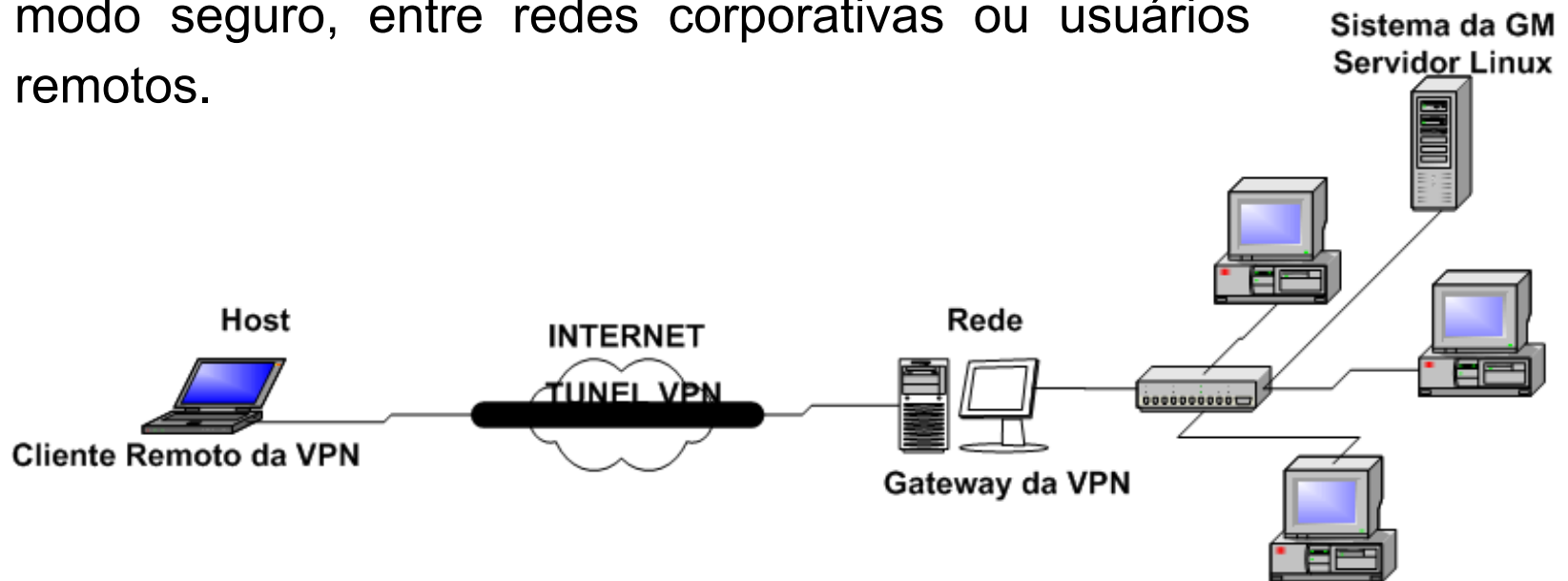
Modelos de Arquitetura

Arquitetura de Sistemas

Cliente-Servidor

Rede Acesso Remoto VPN

- ✓ São túneis de criptografia entre pontos autorizados, criados através da Internet ou outras redes públicas e/ou privadas para transferência de informações, de modo seguro, entre redes corporativas ou usuários remotos.

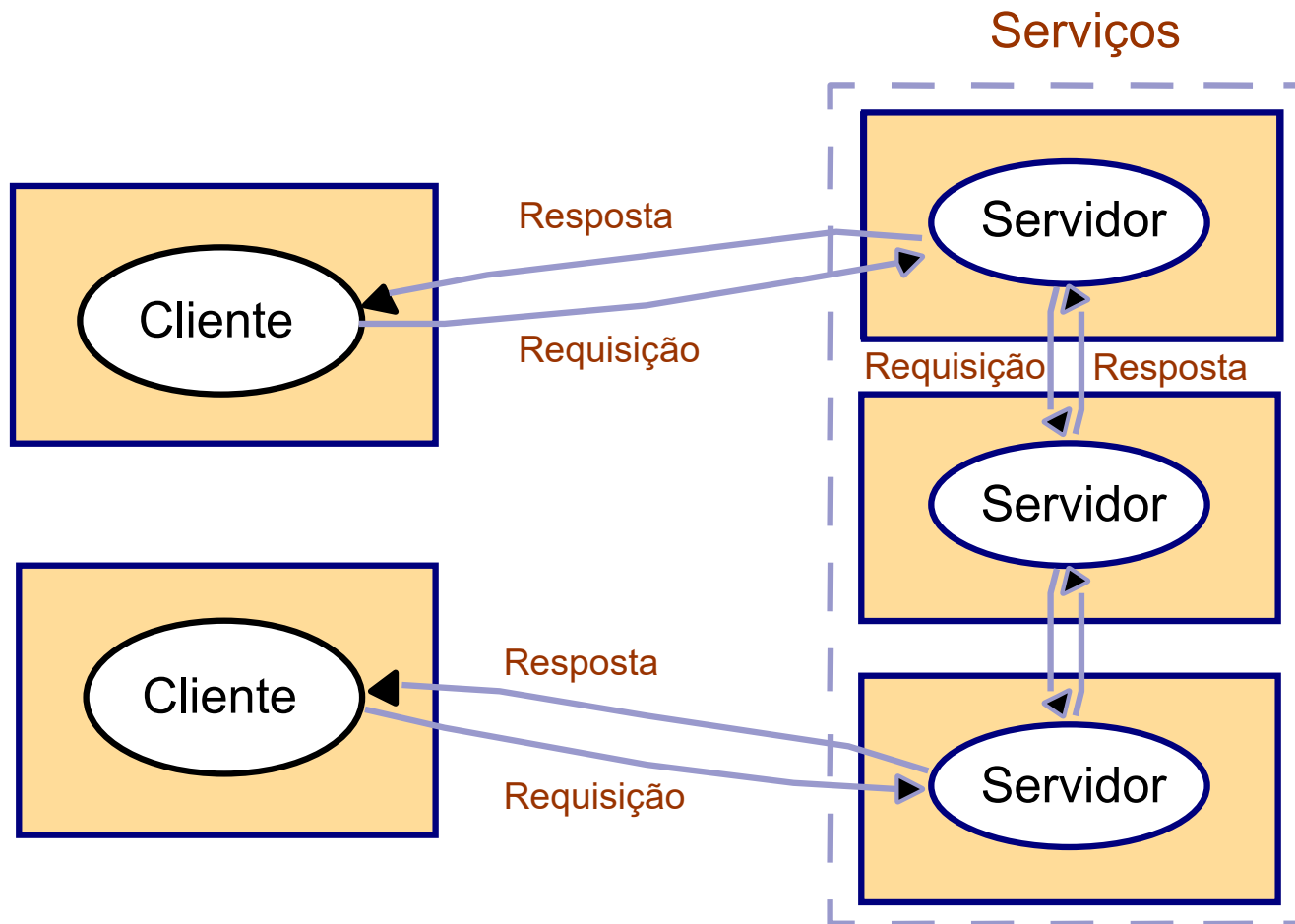


- » Diversos processos servidores em computadores separados interagindo quando necessário para prover um serviço para um processo cliente.
- » O servidor pode particionar o conjunto de objetos no qual o serviço é baseado e distribuir eles no próprio servidor ou pode manter cópias replicadas deles nos diversos servidores em diferentes máquinas.
- » A replicação é usada para aumentar a performance, disponibilidade e aumentar a tolerância a falha.

Modelos de Arquitetura

Arquitetura de Sistemas

Múltiplos Servidores



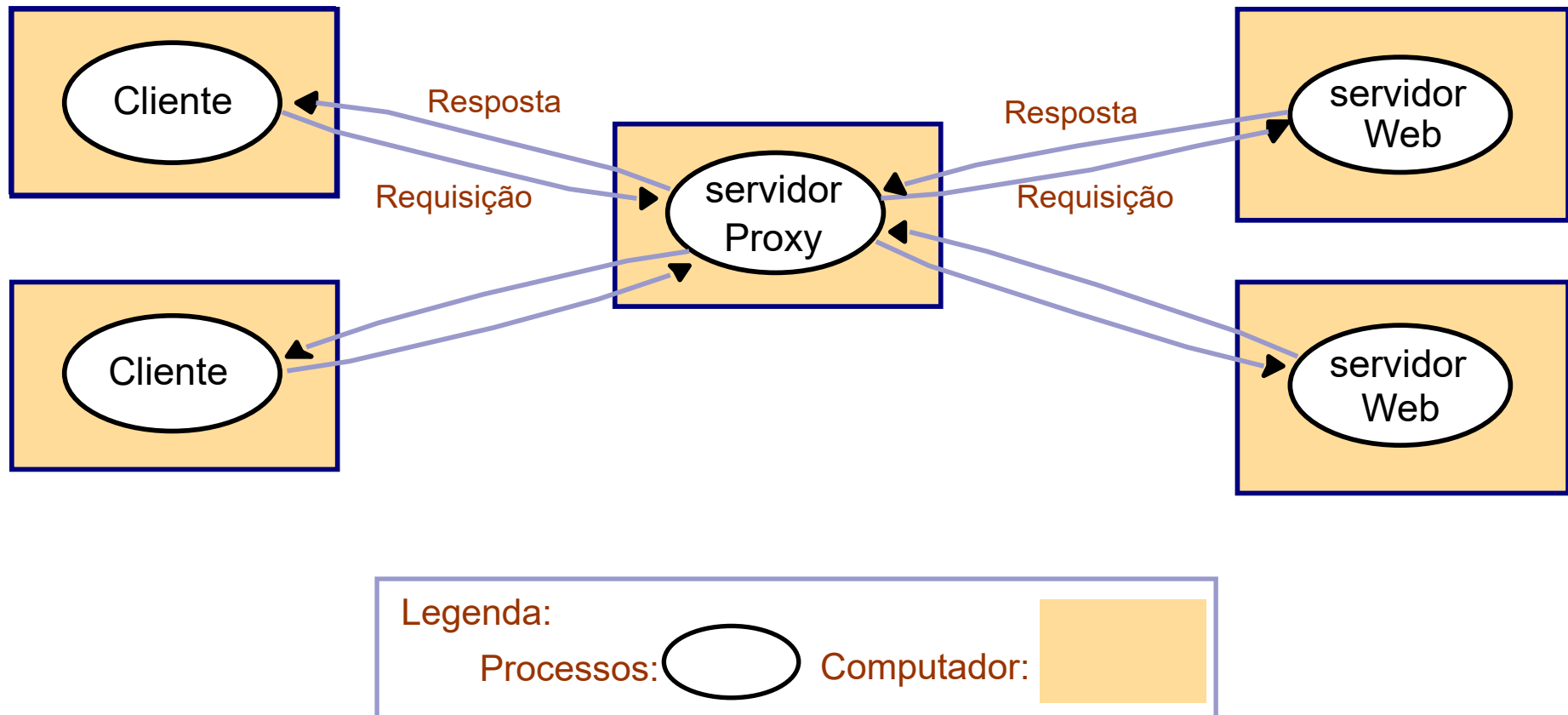
- » Cache é o armazenamento de objetos de dados recentemente usados que estão mais perto do que o próprio objeto de dados;
- » Quando um novo objeto de dado é recebido ele é armazenado no disco do servidor cache, substituindo algum antigo se necessário;
- » Cache pode ser colocado em cada cliente ou em um servidor *proxy* que pode ser compartilhado para diversos clientes;

- » O propósito do servidor *proxy* é aumentar a disponibilidade e desempenho do serviço, reduzindo a carga no tráfego da rede e no servidor web;
- » Servidores proxy podem também ser usados para acessar servidores web remotos através do *firewall*.

Modelos de Arquitetura

Arquitetura de Sistemas

Servidores Caches e Proxy

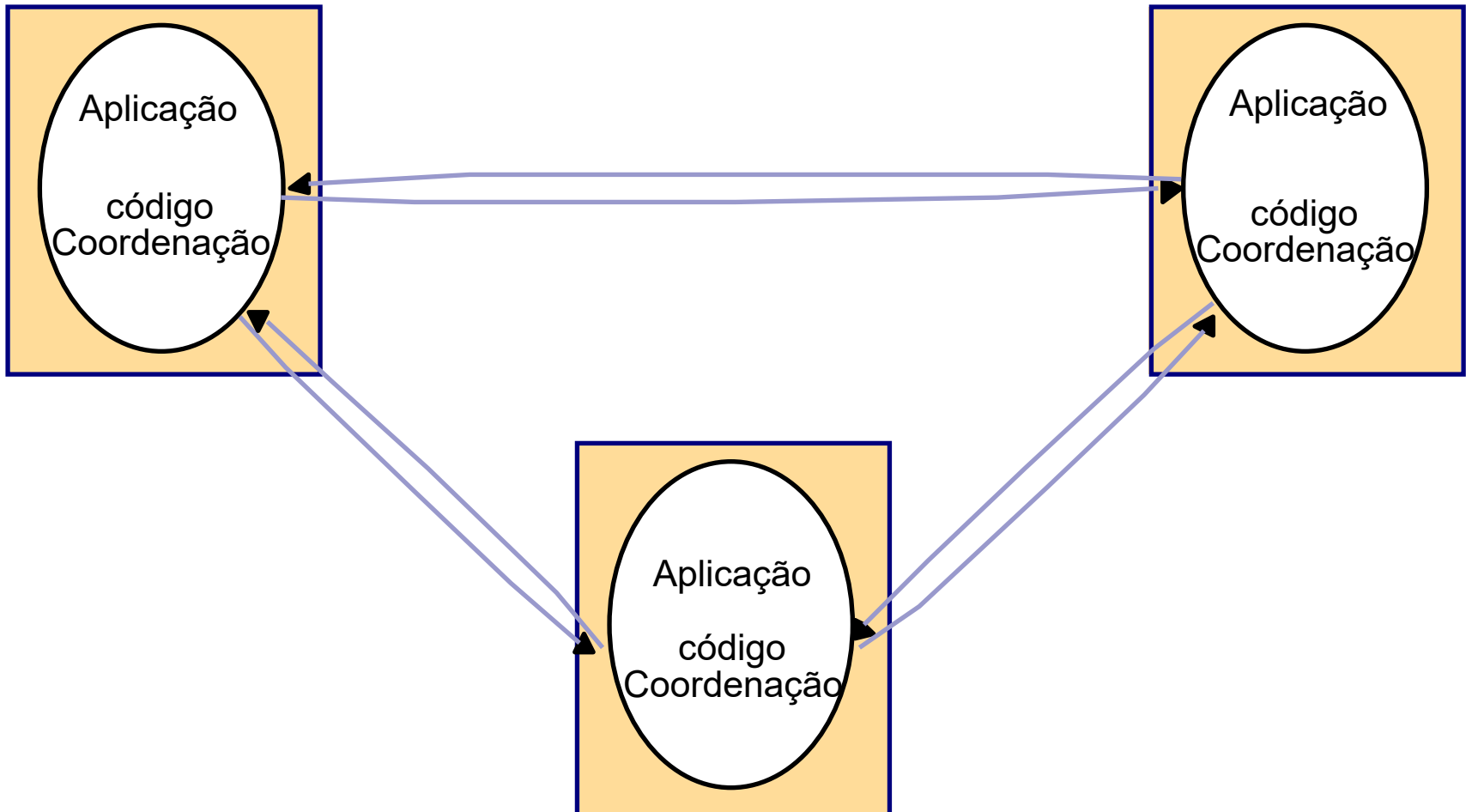


- » Nesta arquitetura todos os processos executam regras similares, interagindo cooperativamente como pares para executar uma atividade ou computação distribuída sem nenhuma distinção entre cliente e servidor.
- » De forma geral, *n* processos *peer* podem interagir entre si e o padrão de comunicação vai depender dos requisitos da aplicação.

Modelos de Arquitetura

Arquitetura de Sistemas

Processos *Peer*

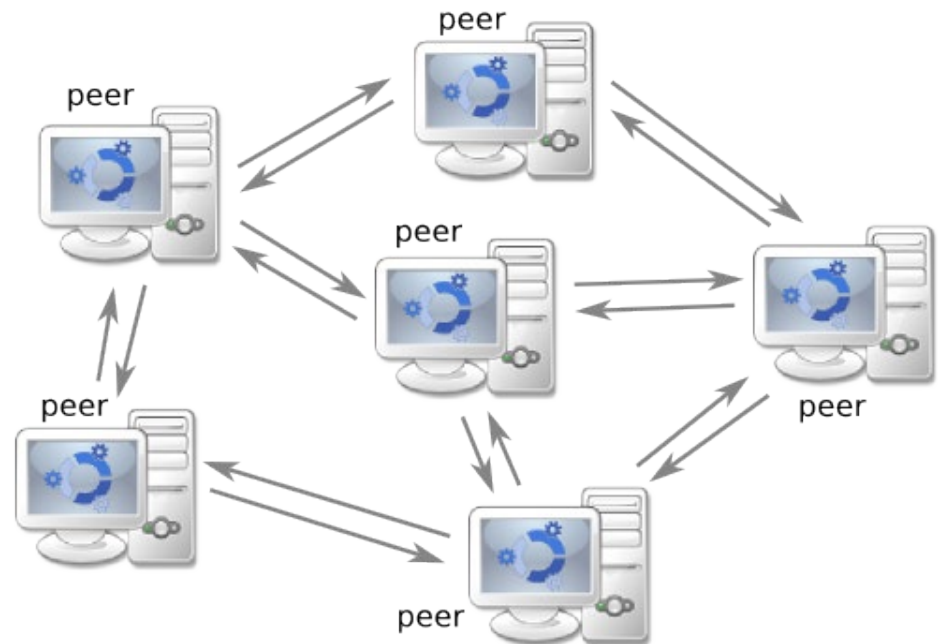


Modelos de Arquitetura

Arquitetura de Sistemas

Processos *Peer*

Sua grande vantagem, em relação à computação cliente/servidor, é possibilitar a colaboração direta entre os usuários, sem depender de servidores administrados por terceiros.



Modelos de Arquitetura

Arquitetura de Sistemas

Processos *Peer*

Compartilhamento de Arquivos



Modelos de Arquitetura

Arquitetura de Sistemas

Mensagem Instantânea

Processos *Peer*

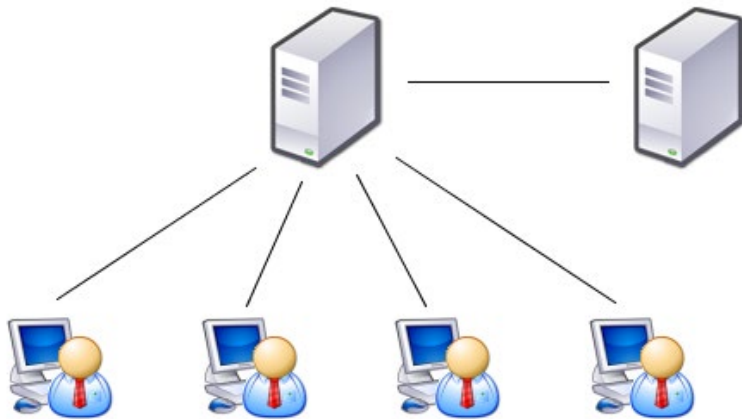


Modelos de Arquitetura

Arquitetura de Sistemas

Resumindo...

- ✓ Estão relacionados com o local onde se encontram as partes e o relacionamento entre elas.



Modelo Cliente-Servidor



Modelo Peer-to-Peer



Modelos de Arquitetura

Arquitetura de Sistemas

Variação do Modelo Cliente-Servidor:

Diversas variações no modelo cliente-servidor podem ocorrer se considerarmos os seguintes fatores:

- » O uso de códigos móveis e agentes móveis;
- » Necessidades do usuário em ter computadores de baixo custo com recursos limitados de hardware e que sejam simples de gerenciar;
- » A necessidade de adicionar e remover dispositivos móveis de forma conveniente.



Modelos de Arquitetura

Arquitetura de Sistemas

Códigos Móveis:

- » *Applets* são exemplos bem conhecidos e largamente usados de código móvel;
- » O usuário executando um *browser* seleciona um *link* para um applet cujo código está armazenado no servidor, este código é recebido no browser e roda nele;

Vantagem:

- ✓ Pode ter um bom tempo de resposta interativa, pois não depende mais dos atrasos e variações na velocidade da rede de comunicação.

Modelos de Arquitetura

Arquitetura de Sistemas

Códigos Móveis

A) Cliente requisita resultados realizando download do código *applet*;



B) Cliente interage com o *applet*;





Modelos de Arquitetura

Arquitetura de Sistemas

Agentes Móveis:

- » É um programa executando que viaja de um computador para outro na rede, carregando uma tarefa para quem possa executar de favor (tal como coleta de informações), eventualmente retornando com o resultado.

Rede de Computadores:

- » É a arquitetura mais conhecida, onde as aplicações rodam em um computador de mesa junto ao usuário. O sistema operacional e os programas aplicativos geralmente estão no disco local.



Modelos de Arquitetura

Arquitetura de Sistemas

Cliente Magro:

- » Refere-se a camada de software que suporta uma interface para o usuário baseado em janelas no computador que é local, enquanto a execução da aplicação é feita em um computador remoto;
- » Semelhantes aos computadores de rede, porém toda a execução é feita em grandes servidores, eventualmente clusters, geralmente multiprocessadores, rodando sistemas como Unix ou Windows NT.

Modelos de Arquitetura

Arquitetura de Sistemas

Cliente Magro

Rede de Computadores
ou PC



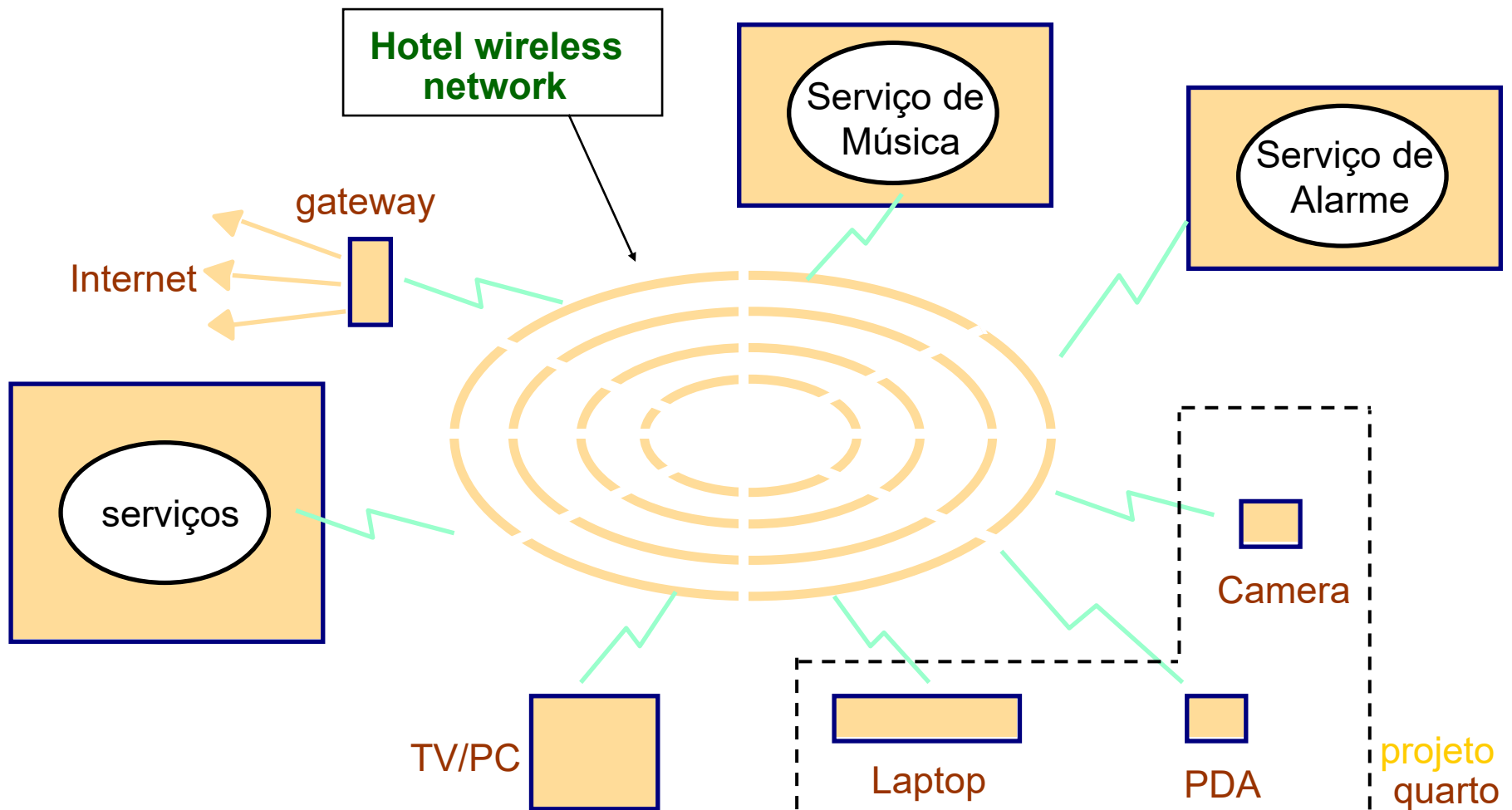
Dispositivos Móveis e Redes Espontâneas

- » A capacidade de se interligar via rede sem fio, com possibilidades de comunicação à distâncias metropolitanas (GSM, CDPD), à distância de centenas de metros (WLAN), ou poucos metros (Infravermelho) é talvez a melhor descrição para o termo redes espontâneas.

Modelos de Arquitetura

Arquitetura de Sistemas

Redes Espontâneas





Modelos Fundamentais

- » Apresentados os Modelos baseados nas propriedades fundamentais que nos permitem ser mais específicos sobre suas características as falhas e os riscos de segurança;
- » Um modelo tem que tratar os seguintes pontos:
 - » Quais as principais entidades dos sistemas?
 - » Como elas interagem?
 - » Quais as características que afetam o seu comportamento individual e coletivo?

Modelos Fundamentais

- Os aspectos dos sistemas distribuídos que desejamos considerar nos modelos fundamentais são:

- **Interação:**

- Processos interagem enviando mensagens uns aos outros, ou seja, a comunicação entre processos

- **Falhas:**

- Definir e classificar falhas

- **Segurança:**

- Definir e classificar as formas com que ataques à segurança do sistemas pode assumir



Modelos Fundamentais

Modelos de Interação

- » Sistemas distribuídos são compostos de muitos processos interagindo entre si de forma complexa;
 - » Múltiplos servidores podem cooperar entre si para prover um serviço;
 - » Um conjunto de processos pares pode cooperar entre si para atingir um objetivo comum.
-
- » Dois fatores significantes que afetam a interação de processos em sistemas distribuídos:



Modelos Fundamentais

Modelos de Interação

Desempenho na comunicação

- A comunicação em uma rede de computadores tem as seguintes características de desempenho:
 - **Latência:** Atraso decorrido entre o início da transmissão de uma mensagem em um processo remetente e o início da recepção pelo processo destinatário.
 - Tempo que o primeiro bit leva para chegar ao seu destino;
 - Atraso no acesso à rede quando está muito carregada;
 - Tempo de processamento gasto pelos serviços de comunicação do sistema operacional nos processos de envio e recepção;



Modelos Fundamentais

Modelos de Interação

Desempenho na Comunicação

- A comunicação em uma rede de computadores tem as seguintes características de desempenho:
 - **Largura de Banda:**
 - Volume total de informações que pode ser transmitido em determinado momento;
 - **Jitter:**
 - Variação no tempo exigida para distribuir uma série de mensagens;
 - Crucial para dados multimídia;



Modelos Fundamentais

Modelos de Interação

Relógios de Computador

- » Cada computador em um sistema distribuído tem o seu próprio relógio (clock);
- » Nem sempre os relógios dos diferentes sistemas marcam um tempo único;
- » Mesmo que os relógios de todos os computadores de um SD fossem inicialmente ajustados com o mesmo horário, com o passar do tempo eles variariam entre si;.



Modelos Fundamentais

Modelos de Interação

Relógios de Computador

» Duas posições podem ser adotadas:

S.D. Síncrono:

- » Todas as operações, processamento e transmissão de mensagens tem limites conhecidos e são controlados por tempo.

S.D. Assíncrono:

- » Alguns sistemas distribuídos não tem limites bem definidos de tempo para velocidade de execução e tempos de transmissão das mensagens.



Modelos Fundamentais

Modelos de Interação

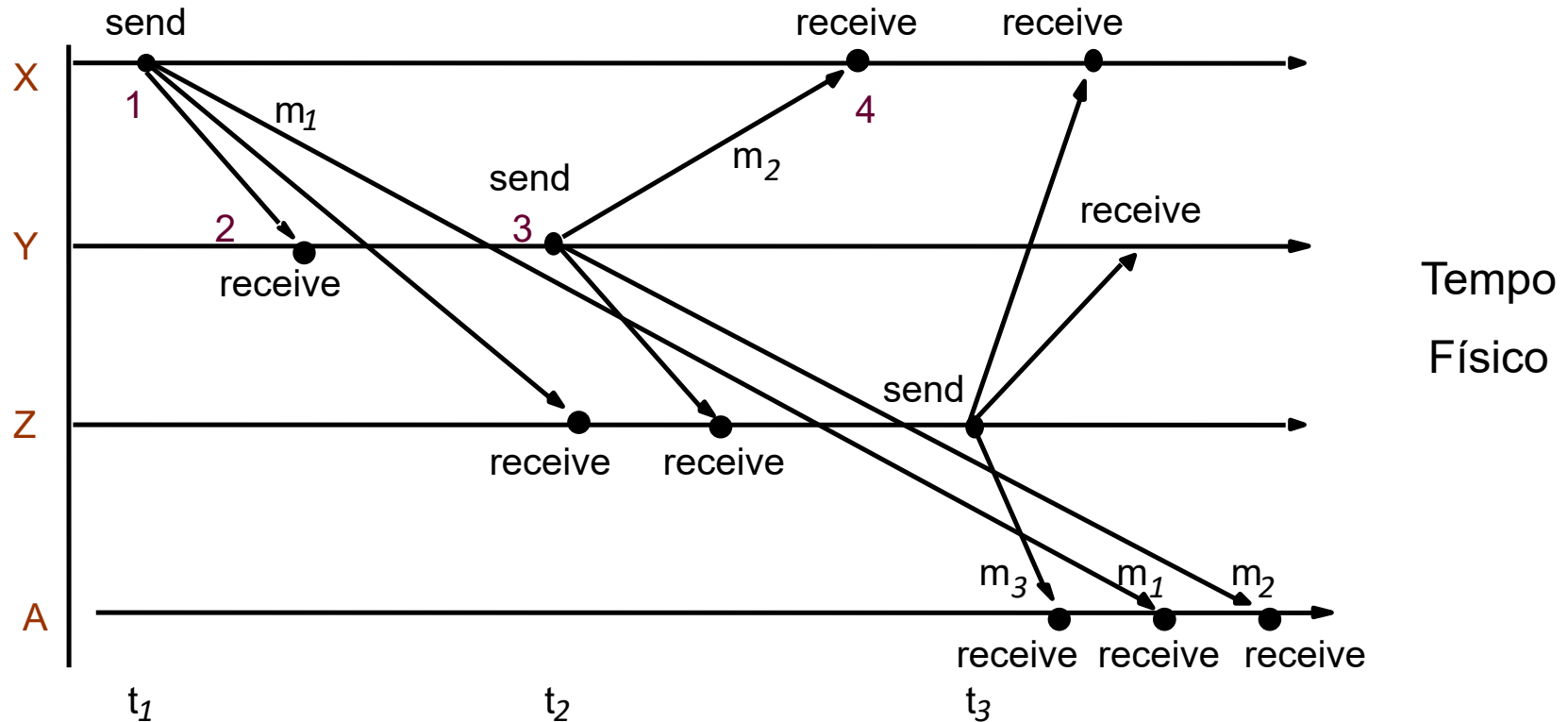
Ordenação de Eventos

- Na ausência da noção de relógio, a execução de um sistema pode ser descrita em termo da ocorrência de eventos e sua ordem.
- Exemplo
 - O usuário X envia uma mensagem com o assunto “Reunião”;
 - Os usuários Y e Z respondem, enviado uma mensagem com o assunto “Re:Reunião”.

Modelos Fundamentais

Modelos de Interação

Ordenação de Eventos (Relógios de Computador)





Modelos Fundamentais

Modelo de Falhas

- ✓ Em um sistema distribuído, tanto os processos como os canais de comunicação podem falhar.
- ✓ Podemos ter:
 - **FALHAS DE OMISSÃO:** São falhas nos casos onde um processo ou canal de comunicação falha na execução da ação que deveria fazer.
 - **Processo:**
 - Ocorre quando o processo entra em colapso, parando e não executando outro passo de seu programa.
 - Falhas pode ser detectadas por *timeouts*.



Modelos Fundamentais

Modelo de Falhas

Falhas de Omissão:

- Parada por falha:

- **Sistemas Assíncronos:**

- O *timeout* indica que um processo não está respondendo, sem entrar em detalhes;
 - Colapso, Estar Lento, Mensagens não chegaram;

- **Sistemas Síncronos:**

- O *timeout* é usado para determinar que processos deixaram de responder a mensagens que foram entregues;



Modelos Fundamentais

Modelo de Falhas

Falhas de Omissão:

- **Canal de Comunicação:**

- O canal de comunicação produz uma falha por omissão quando não concretiza a transferência de uma mensagem.
- Pode ser causado por:
 - Falta de espaço no buffer de recepção;
 - Erro durante a transmissão gerando uma mensagem inválida, que é descartada.
 - Cálculo da paridade da mensagem;

Modelos Fundamentais

Modelo de Falhas

FALHAS DE OMISSÃO: Canal de Comunicação





Modelos Fundamentais

Modelo de Falhas

- Canal de Comunicação:

- Falhas por omissão de envio:

- Perda de mensagens entre o processo remetente e o buffer de envio.

- Falhas por omissão de recepção:

- Perda de mensagens entre o buffer de recepção e o processo de destino.

- Falhas por omissão de canal:

- Perda de mensagens no meio de comunicação



Modelos Fundamentais

Modelo de Falhas

- ✓ **FALHAS ARBITRÁRIAS:** O termo é empregado para descrever falhas de semântica, ou seja, qualquer tipo de erro pode ocorrer.
 - » Um processo pode colocar valores errados nos seus itens de dados;
 - » Um processo pode retornar valores errados em resposta a uma chamada;
 - » Uma solicitação que não seja mais necessária pode ser entregue;
 - » Uma mensagem real pode ser entregue mais de uma vez.

Modelos Fundamentais

Modelo de Falhas por Omissão e Arbitrárias

<i>Tipos de falhas</i>	<i>Afeta</i>	<i>Descrição</i>
Parada por falha	Processo	O processo pára e permanece parado. Outros processos podem detectar esse estado.
Colapso	Processo	O processo pára e permanece parado. Outros processos podem não detectar esse estado.
Omissão	Canal	Uma mensagem inserida em um buffer de envio nunca chega no buffer de recepção do destinatário.
Omissão de envio	Processo	Um processo conclui o envio, mas a mensagem não é colocada em seu buffer de envio.
Omissão de recepção	Processo	Uma mensagem é colocada no buffer de recepção de um processo, mas esse processo não recebe a mensagem.
Arbitrária (Bizantina)	Processo Ou canal	O processo/canal exibe comportamento arbitrário: ele pode enviar/transmitir mensagens arbitrárias em qualquer momento, cometer omissões; um processo pode parar ou realizar uma ação incorreta.

Modelos Fundamentais

Modelo de Falhas

- ✓ **FALHAS DE TEMPORIZAÇÃO:** Falhas de tempo são aplicáveis a sistemas distribuídos síncronos, onde limites de tempo são estabelecidos para tempo de execução do processo, tempo de entrega da mensagem e a taxa de erro do relógio é conhecida.

<i>Classe da Falha</i>	<i>Afeta</i>	<i>Descrição</i>
Relógio	Processo	O relógio local do processo ultrapassa os limites de sua taxa de desvio em relação ao tempo físico.
Desempenho	Processo	O processo ultrapassa os limites do intervalo de tempo entre duas etapas.
Desempenho	Canal	A transmissão de uma mensagem demora mais do que o limite definido.



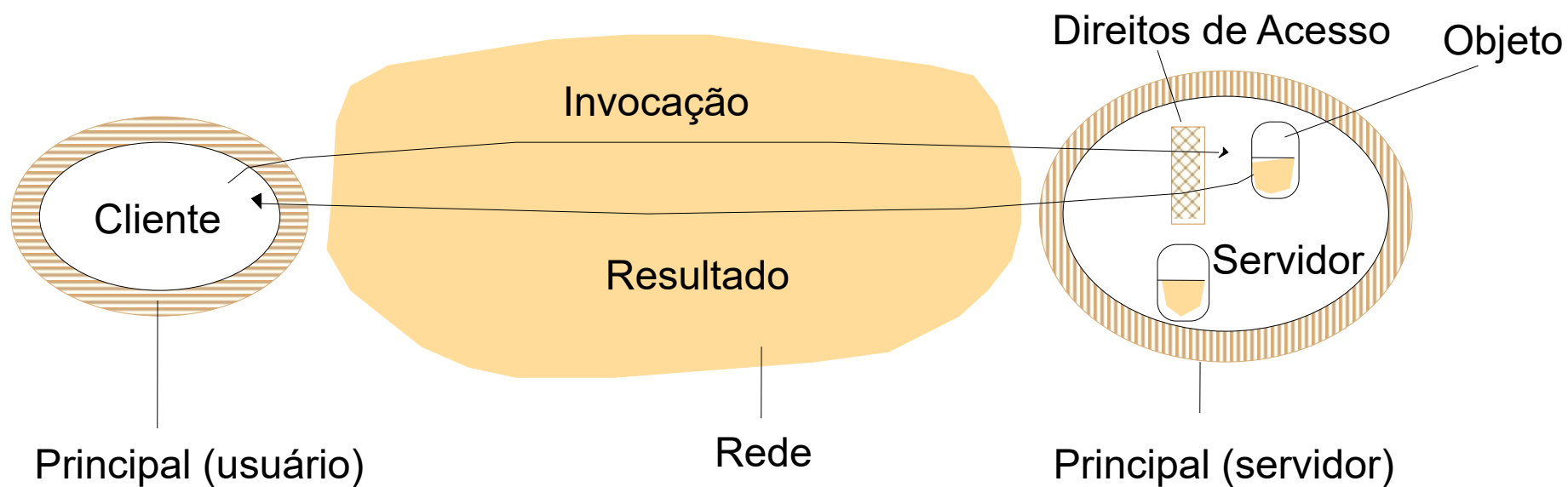
Modelos Fundamentais

Modelo de Segurança

- A segurança de um sistema distribuído pode ser obtida tornando seguros os processos e os canais usados por suas interações e protegendo contra acesso não autorizado os objetos que encapsulam.
- Direitos de acesso especificam quem pode executar determinadas operações sobre um objeto.
 - Quem pode ler ou gravar em seus atributos.
- Os usuários devem ser incluídos no modelo de segurança como beneficiários dos direitos de acesso.

Modelos Fundamentais

Modelo de Segurança





Modelos Fundamentais

Modelo de Segurança

Processos e Interações Seguros

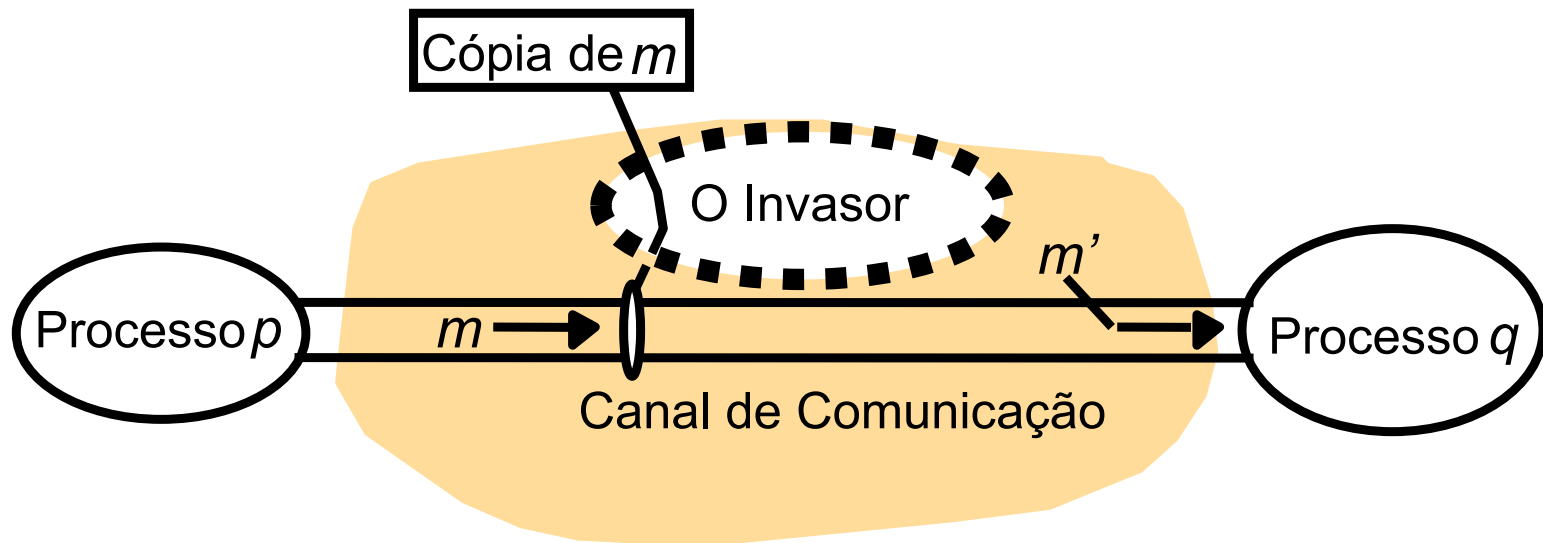
- No modelo, um invasor poderá ser capaz de enviar qualquer mensagem para qualquer processo e ler ou copiar qualquer mensagem entre dois processos.
- Processos interagem trocando mensagens que ficam expostas ao ataque.
- Servidores e processos *peer-to-peer* publicam suas interfaces, permitindo que invocações sejam enviadas a eles por qualquer outro processo.

Modelos Fundamentais

Modelo de Segurança

Processos e Interações Seguros

- O INVASOR: Capaz de enviar ou ler qualquer mensagem de qualquer processo;
- O ataque pode vir de um computador legítimo da rede, ou de um não autorizado.





Modelos Fundamentais

Modelo de Segurança

Processos e Interações Seguros

- Ameaças aos processos:
 - Servidores não podem necessariamente confirmar a identidade do usuário por trás da solicitação.
 - Clientes não podem garantir que a resposta a uma solicitação feita a um servidor é ou não a resposta de um invasor ou do servidor.
 - **IP Spoofing** é uma técnica de subversão de sistemas em rede que consiste em mascarar (*spoof*) pacotes IP utilizando endereços de remetentes falsificados.



Modelos Fundamentais

Modelo de Segurança

Processos e Interações Seguros

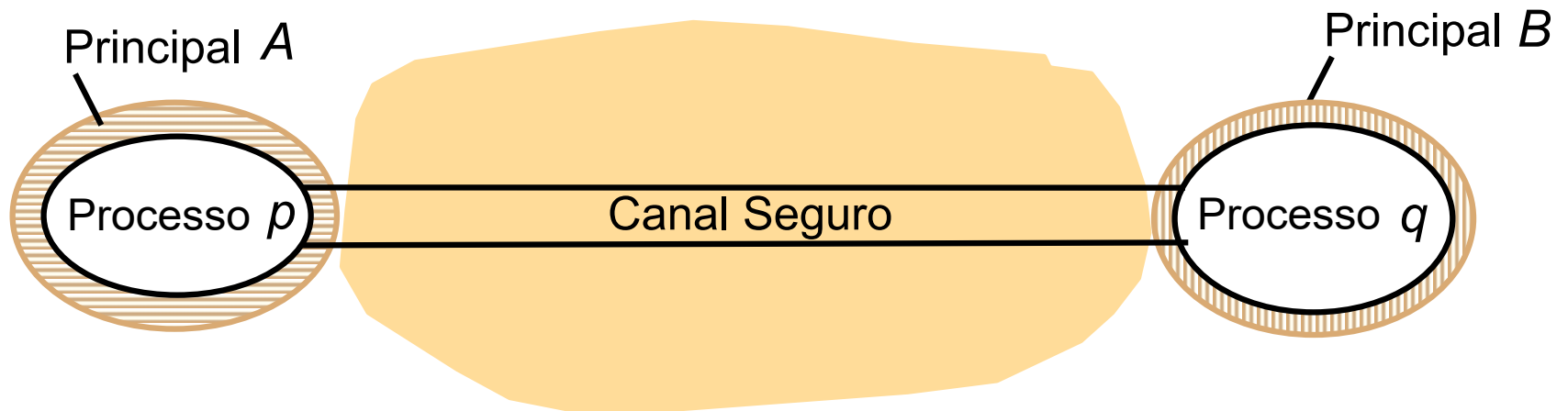
- Ameaças aos canais de comunicação:
 - Invasores podem copiar, alterar ou injetar mensagens quando elas trafegam pela rede e em seus sistemas intermediários.
 - Ameaça à integridade;
 - Ameaça à privacidade;
 - Ameaça ao sistema;
 - Invasores podem salvar mensagens de solicitação de serviços e depois reproduzi-las.

Modelos Fundamentais

Modelo de Segurança

Processos e Interações Seguros

- Criptografia;
- Autenticação;
- Canais seguros;





Resumindo...

- A maioria dos sistemas distribuídos é organizada com base em um, entre vários, modelos de arquitetura de sistemas, predominando o modelo cliente-servidor;
- No modelo *peer-to-peer* todos os processos desempenham funções semelhantes na exploração dos recursos disponibilizados;
- A capacidade de mover código de um processo para outro tem resultado em algumas variantes do modelo cliente-servidor;
- Os modelos de interação, falha e segurança identificam as características comuns dos componentes básicos a partir dos quais os SD são construídos.

Atividade I em Grupo

- ↪ Realizar o download do Cisco Packet Tracer ;
 - ↪ <https://www.netacad.com/pt-br/courses/packet-tracer>
- ↪ Projetar uma rede para oferecer serviços WEB/FTP/DNS – utilizar mínimo 3 nós (Roteadores);
- ↪ A entrega será realizada através de exposição on-line;
- ↪ A atividade será realizada em grupos já definidos;
- ↪ Cada grupo descreverá, em um relatório técnico a atividade desenvolvida.
- ↪ O relatório será enviado para o e-mail: rviegas@ufpa

Atividade I em Grupo

- ↩ Configurar no simulador:
 - ↩ Topologia da rede;
 - ↩ Tipo de comunicação (links);
 - ↩ Largura de banda;
 - ↩ Atraso Padrão;
 - ↩ Protocolos: ICMP, TCP ou UDP;
 - ↩ Tipo de Serviço: WEB/FTP/DNS;