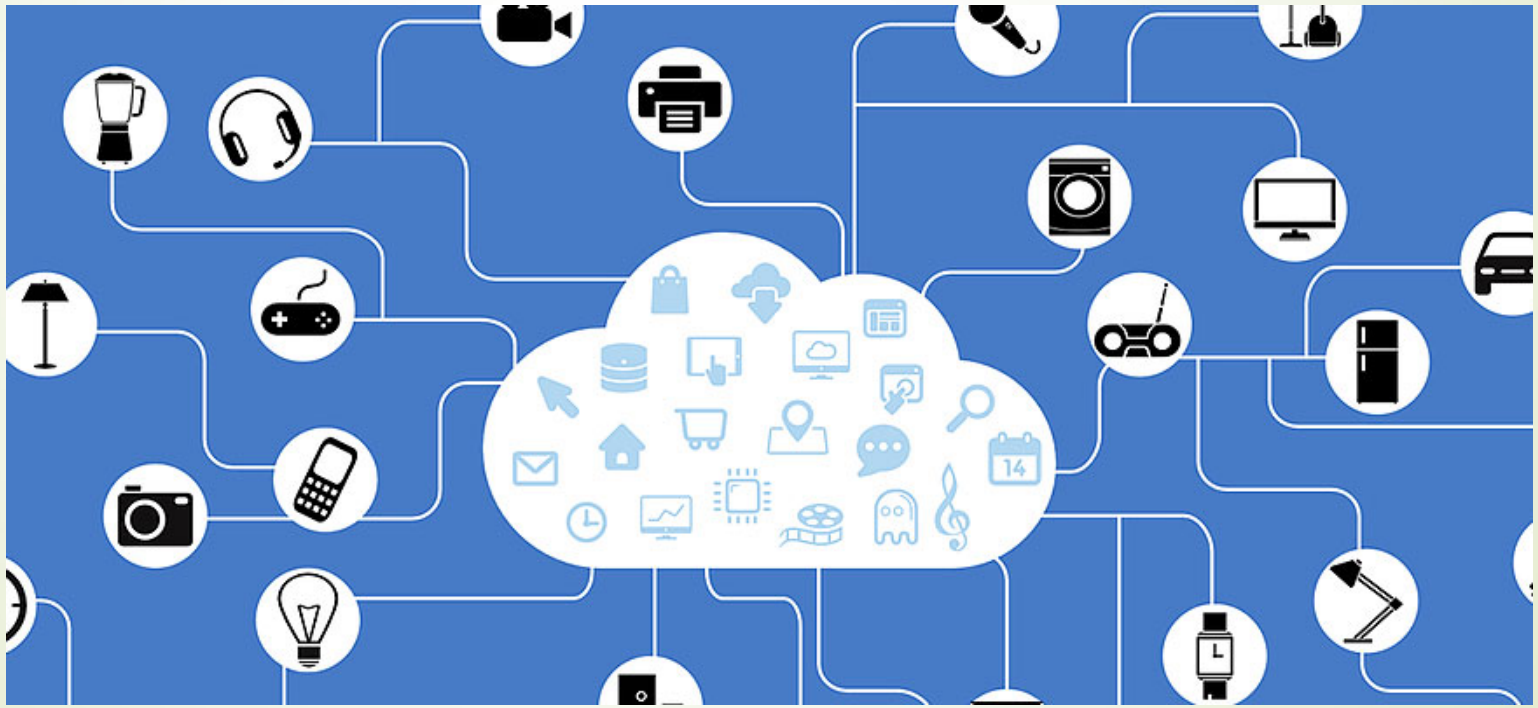
SMD0050 - SISTEMAS DISTRIBUÍDOS

Introdução a Sistemas Distribuídos

Slides são baseados nos slides do Couloris e Tanenbaum

O que são Sistemas Distribuídos?

- Um sistema distribuído é um conjunto de computadores ou entidades computacionais independentes que se apresenta a seus usuários como um sistema único e coerente.



Andrew S. Tanenbaum
Maarten Van Steen

Principais características

- Diferenças entre os vários computadores e o modo como eles se comunicam estão, em grande parte, ocultas ao usuário;
- Usuários e aplicações podem interagir com um sistema distribuído de maneira consistente e uniforme, independentemente de onde a interação ocorra
- Coordenação entre as diversas partes do sistema para executar uma tarefa

Andrew S. Tanenbaum
Maarten Van Steen

Metas de um Sistema Distribuído
Acesso a recursos

- Objetivo: facilitar acesso e compartilhamento de recursos remotos de maneira controlada e eficiente
  - WEB
  - Sistemas colaborativos
- Necessário, no entanto, aprimorar a segurança, evitando acessos indevidos e rastreamento de comunicações e para criação de perfis de usuários.

Andrew S. Tanenbaum
Maarten Van Steen

Uso de Middleware para sistemas heterogêneos

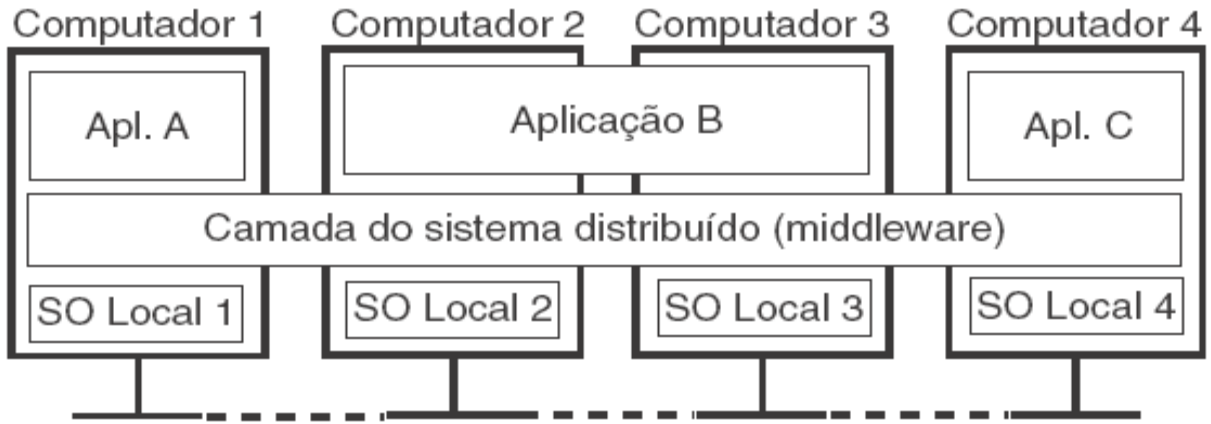


Figura 1.1 *Sistema distribuído organizado como middleware. A camada de middleware se estende por várias máquinas e oferece a mesma interface a cada aplicação.*

Andrew S. Tanenbaum
Maarten Van Steen

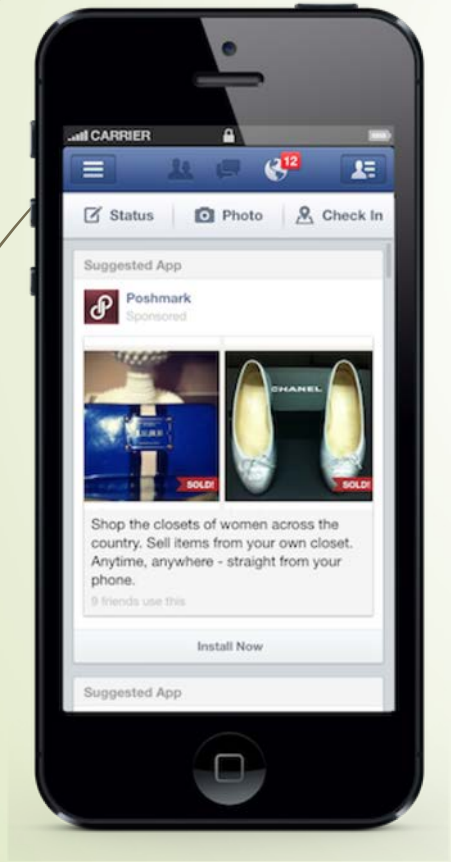
Exercício

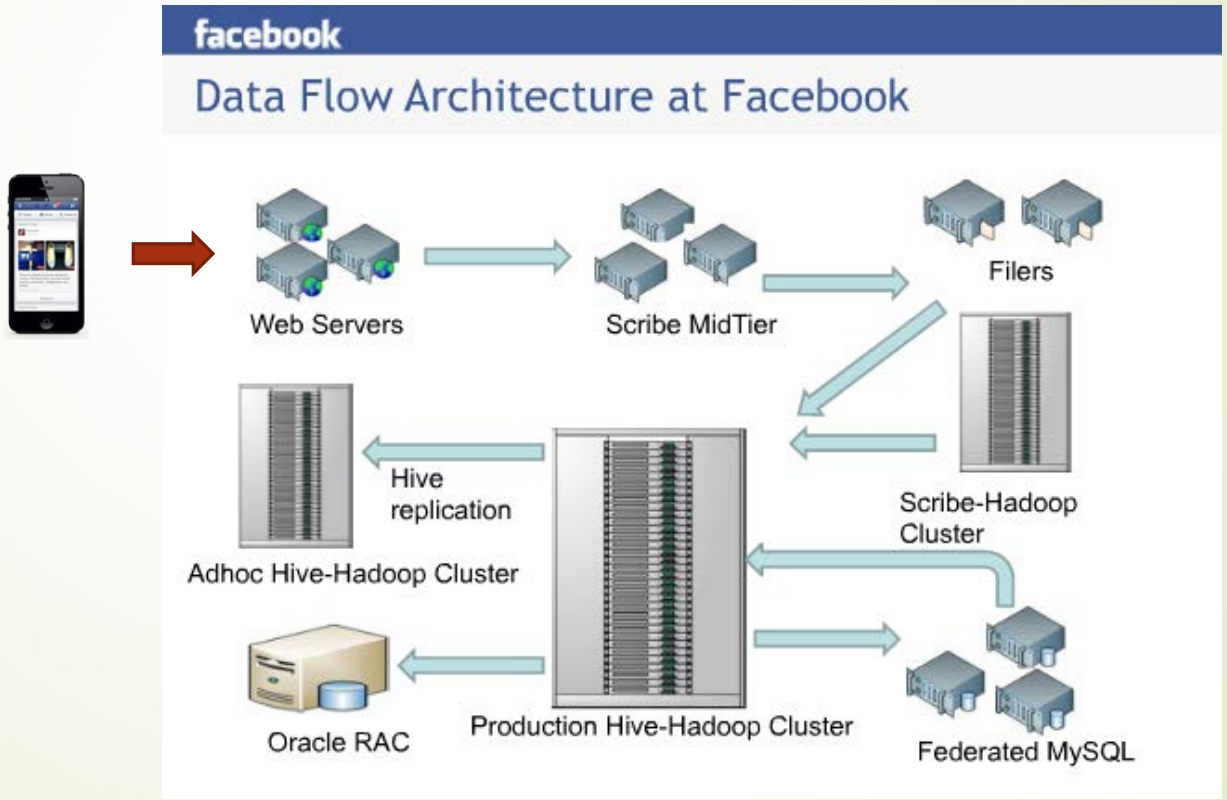
- Cite 3 exemplos de Sistemas Distribuídos a partir da definição apresentada

- Cite 3 desafios na implementação destes sistemas e porque eles são complexos de serem implementados

Exemplos de Sistemas Distribuídos

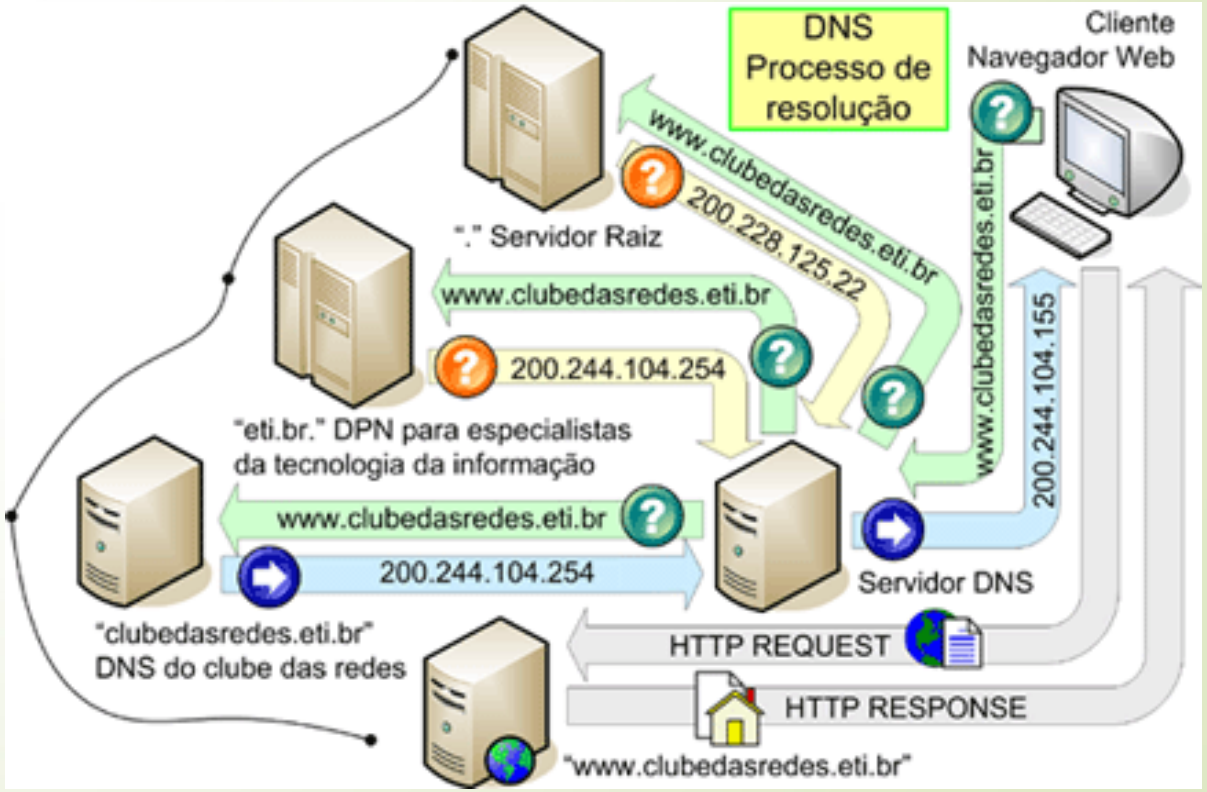
- Sistemas Web
  - Aplicativos móveis que acessam informações da Web





Exemplos de Sistemas Distribuídos

- DNS
  - Uma hierarquia de servidores que cooperam para prover um mapeamento de nome/ip de um host



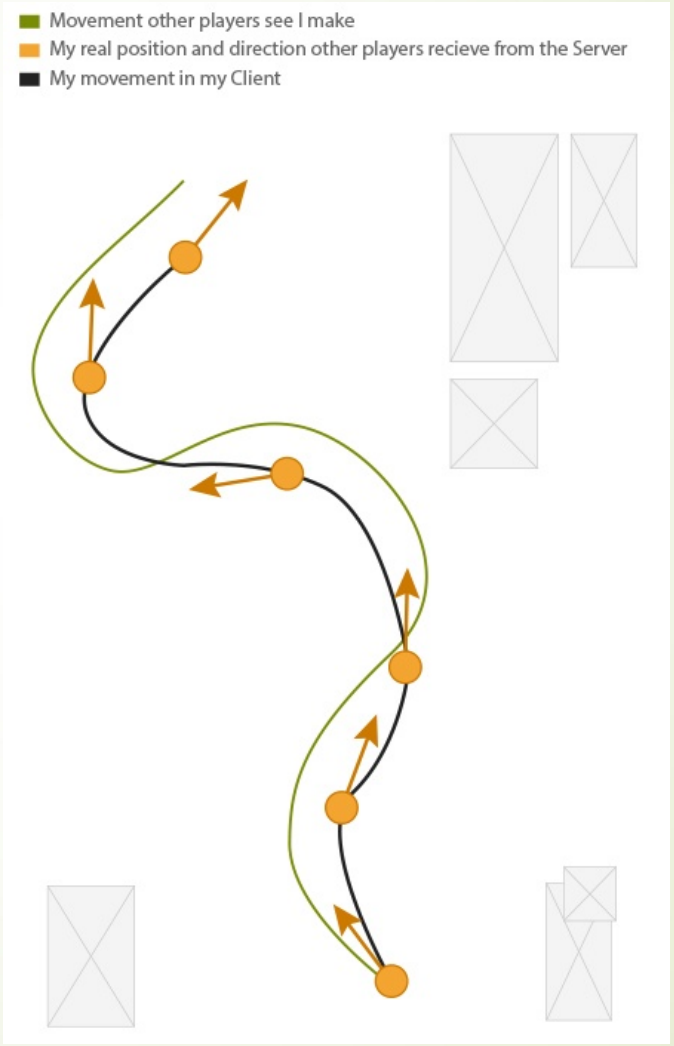
Exemplos de Sistemas Distribuídos

- Jogos Multiusuários "Online"
  - Sincronismo do estado compartilhado do jogo em vários nós
  - Escalável para tipos "massivos"

Fig. 1. (a) Different c Xienzle et al. 2009]
(adapted, ©ACM 2009) ed. The components
discussed in this article Players can interact
with each other, objects

Exemplos de Sistemas Distribuídos

- Jogos Multiusuários "Online"
  - Lag Compesation
  - Dead Reckoning Algorithm



Exemplos de Sistemas Distribuídos

- Drones que cooperam
  - https://www.youtube.com/watch?v=i3ernrkZ91E
- Computação em Nuvem
  - Múltiplas réplicas de servidores coordenados funcionando para o usuário como um único sistema
  - https://www.youtube.com/watch?v=ae_DKNwK_ms
- Seti@Home
  - Buscando Ets de forma distribuída desde 1999
  - https://www.youtube.com/watch?v=_aIJV5aQR68
- Blockchain
  - https://www.youtube.com/watch?v=SSo_EIwHSd4

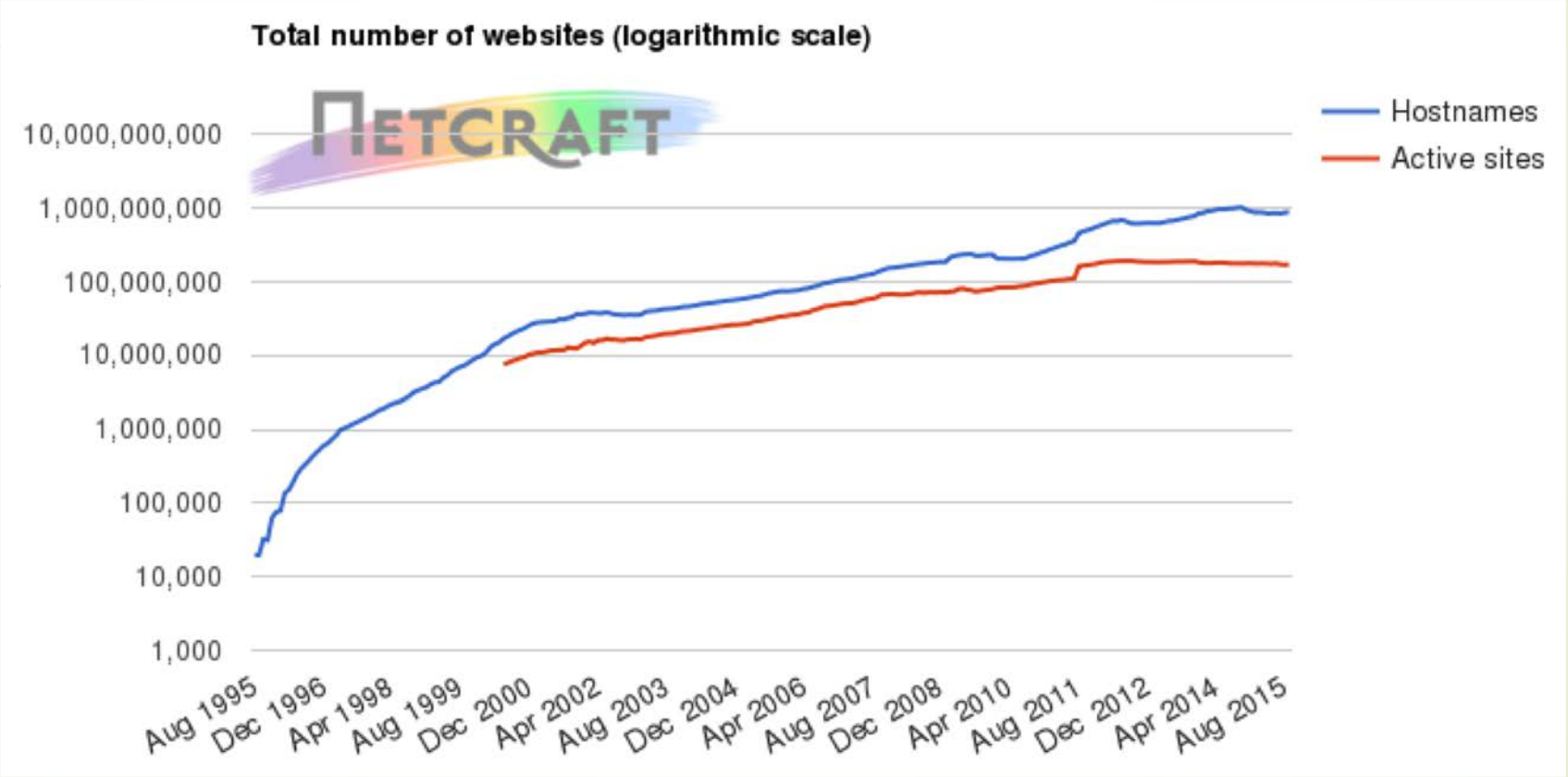
A WEB

- Aplicações e protocolos executando na Web são o típicos exemplos de SD
  - HTTP, DNS, SMTP, Browser
- Qual é o grande objetivo?
  - Compartilhamento de recursos
- Modelo arquitetural cliente-servidor
  - Cliente : invoca uma operação no servidor remoto
  - Servidor: responde a invocação com um recurso ou resultado de um método
  - Múltiplos clientes!

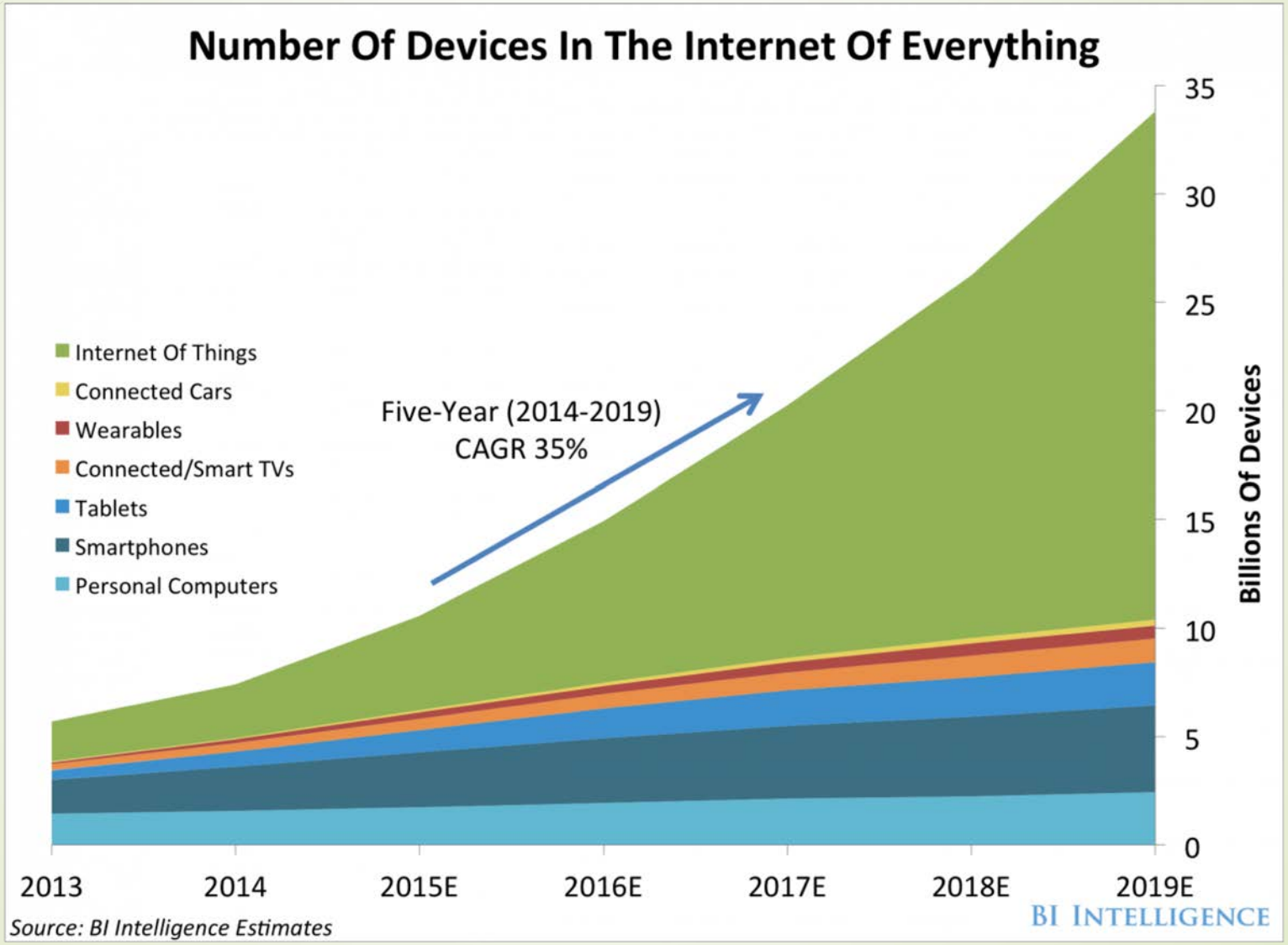
Quão grande é o número de servidores?

Date	*Computers*	*Web servers*	*Percentage*
1993, July	1,776,000	130	0.008
1995, July	6,642,000	23,500	0.4
1997, July	19,540,000	1,203,096	6
1999, July	56,218,000	6,598,697	12
2001, July	125,888,197	31,299,592	25
2003, July	~200,000,000	42,298,371	21
2005, July	353,284,187	67,571,581	19

Quão grande é o número de servidores?



E clientes da Web?

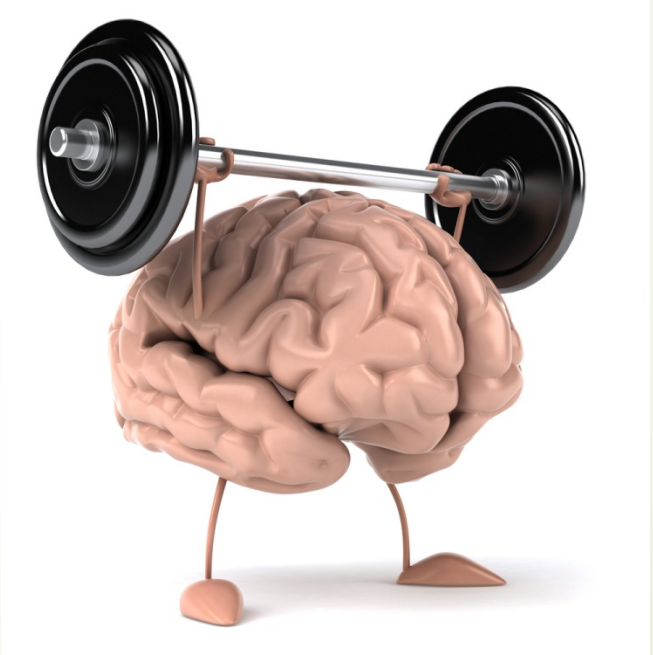


Desafios de SD

I. Heterogeneidade
II. Abertura (openess)
III. Segurança
IV. Escalabilidade
V. Tratamento e Recuperação de Falhas
VI. Concorrência
VII. Transparência

Exercício

- Defina cada um dos desafios e tente relacioná-los com os sistemas distribuídos apresentados



I- Heterogeneidade

- Múltiplos dispositivos, plataformas, modelos de codificação dos dados, tipos de navegadores, condições de acesso
- Soluções
  - Máquinas Virtuais
  - Standards
  - Middlewares
  - Mecanismos de Adaptação

Middleware

- Middleware mascara heterogeneidade de sistemas distribuídos
  - Java VM: SO
  - CORBA: Linguagem
  - OSGi
- Trata de concorrência, portabilidade, interoperabilidade
  - API única, protocolo comum, conjunto de serviços
- Middlewares tradicionais facilitam a vida do desenvolvedor pela abstração de transparência.

Integração usando Middleware de comunicação

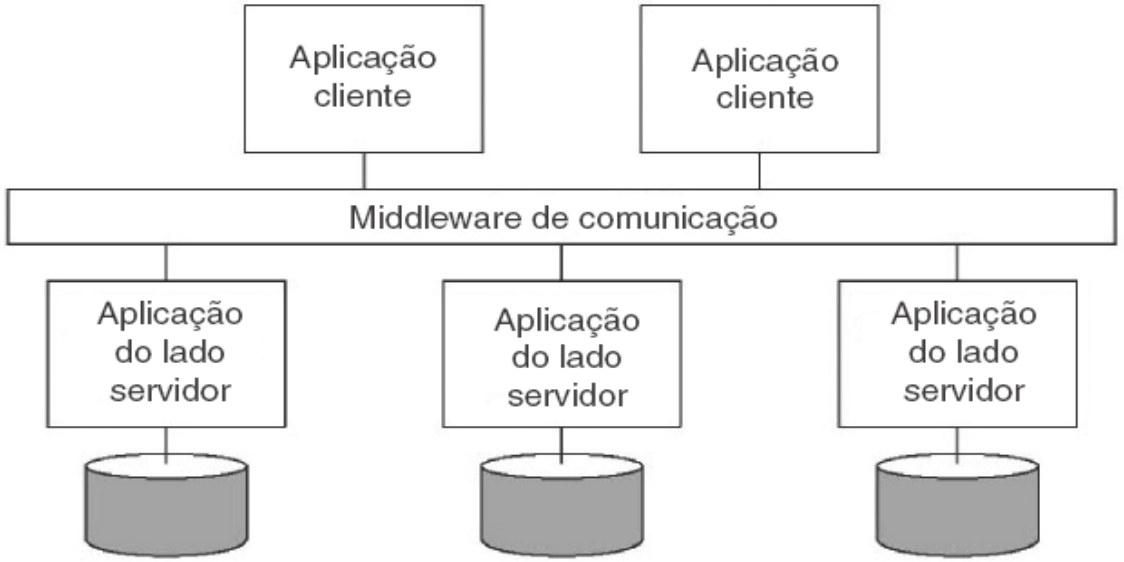


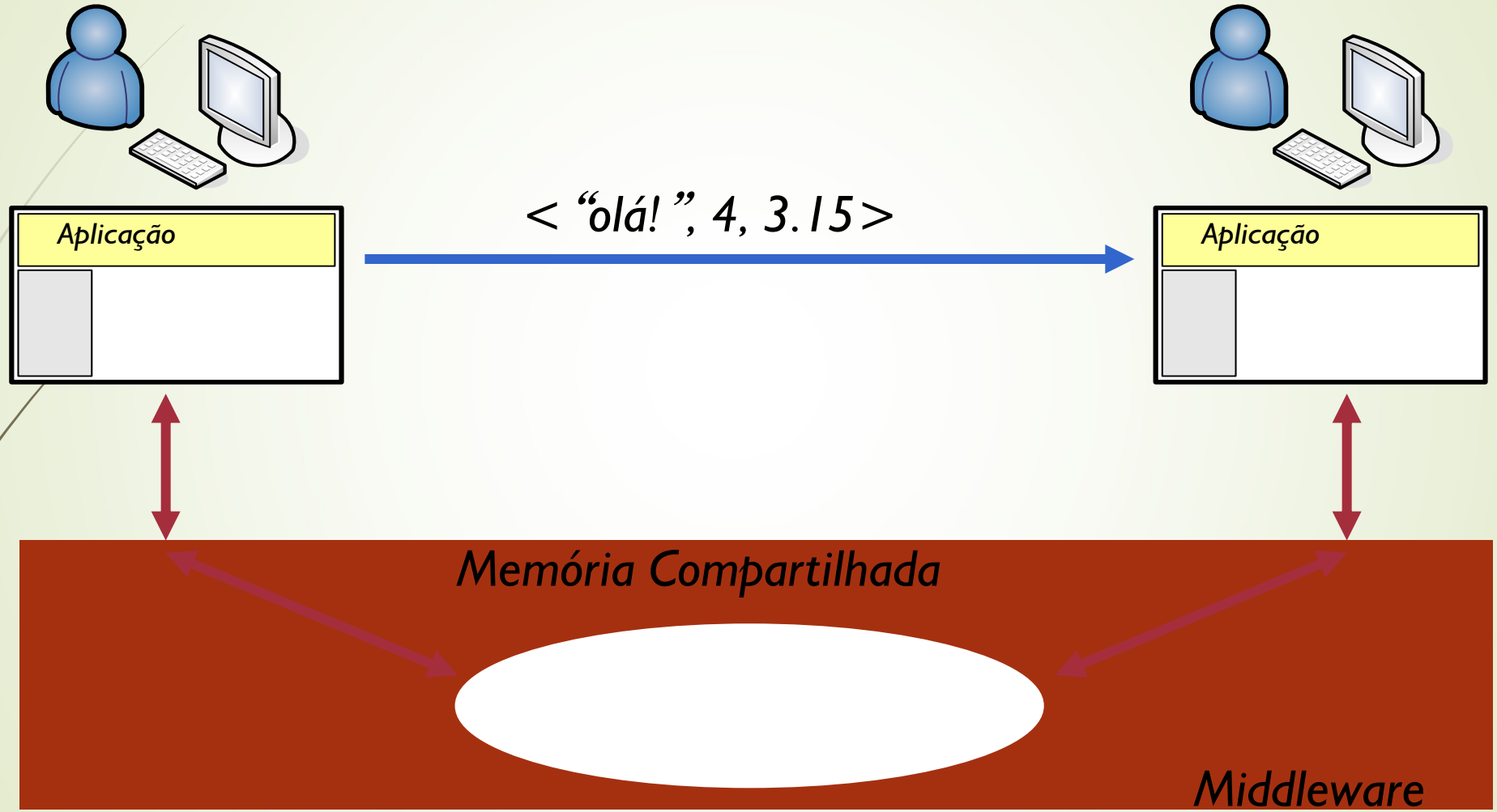
Figura 1.8 Middleware como facilitador de comunicação em integração de aplicações empresariais.

Andrew S. Tanenbaum
Maarten Van Steen

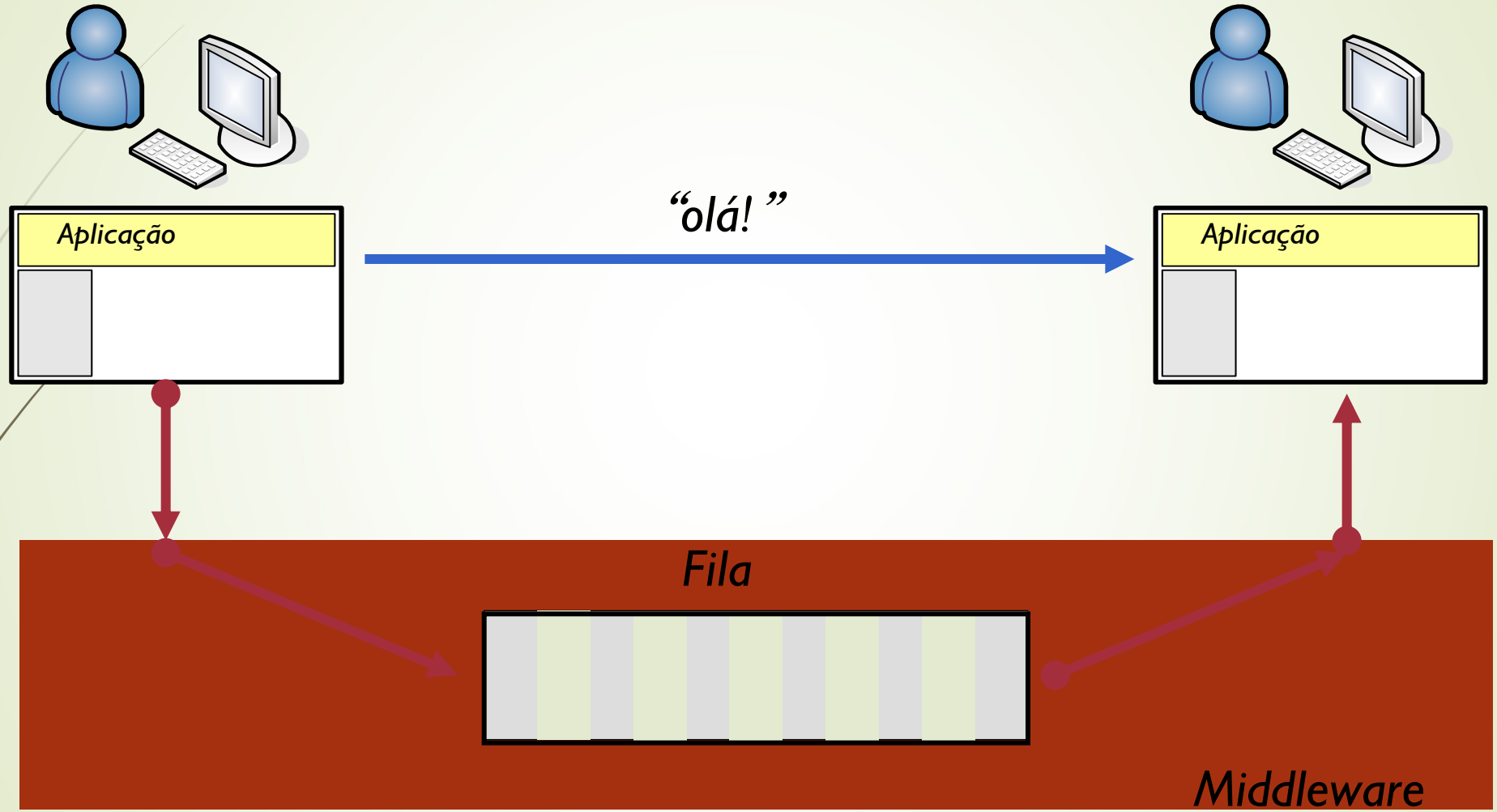
Modelos de Middleware Tradicionais

- Middleware orientados a transação
- Middleware com memória compartilhada
- Middleware orientados a mensagem
- Middleware orientados a RPC
- Middleware orientados a Objetos
- Middleware Adaptativo
- Middleware Multimídia

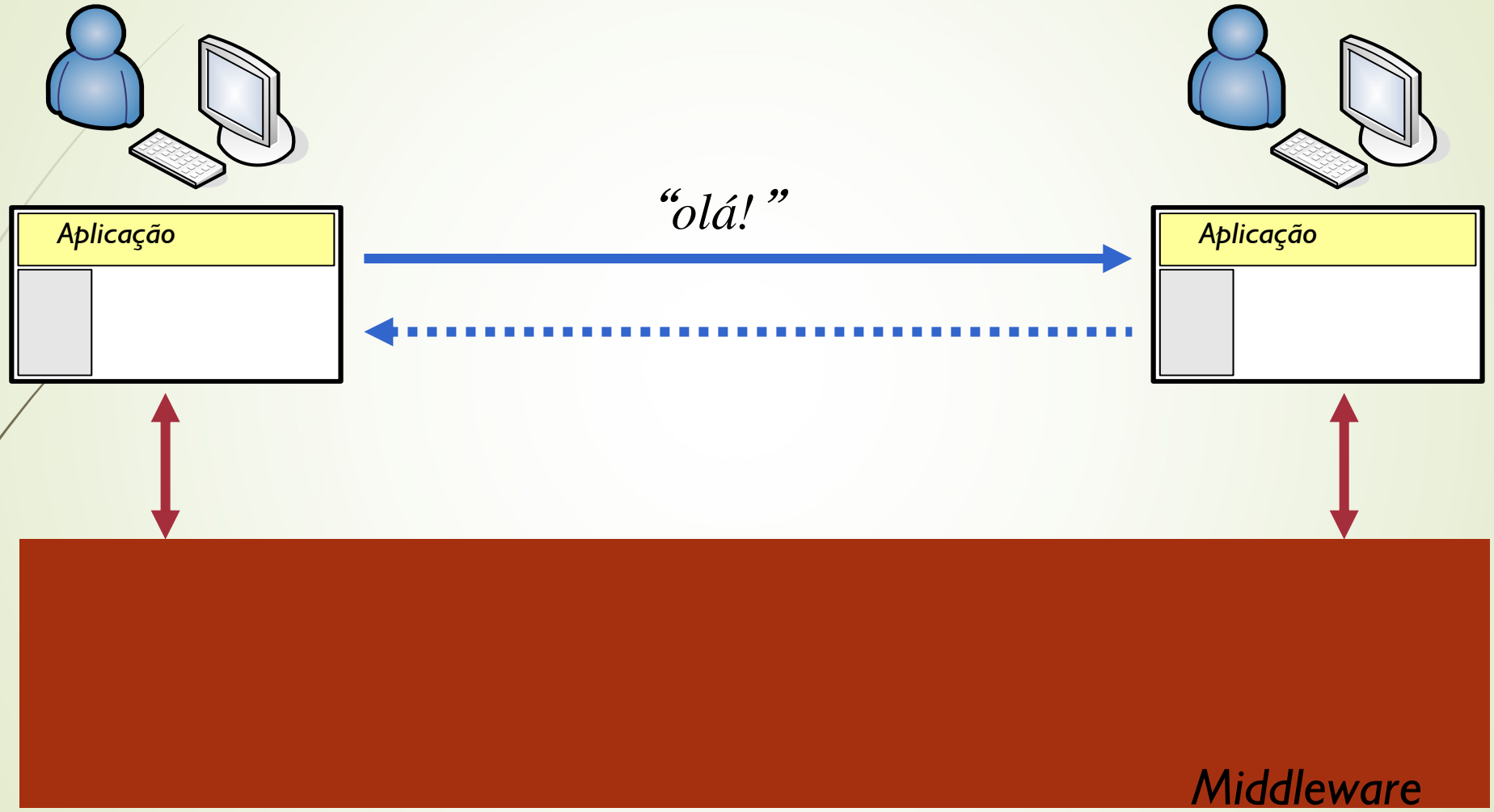
Memória Compartilhada



Orientado a Mensagem



RPC/Objetos



II-Abertura ou Sistemas Abertos

- Sistemas capazes de serem acessados, integrados e modificados facilmente por terceiros
  - Não dependem de um único vendedor ou proprietário para serem utilizados
- Exige o uso de Padrões (Standards) abertos
  - HTTP, XML, WSDL, HTML, SOAP, JSON
  - WIFI, Bluetooth, DLNA, UPNP
  - OSGi

Metas de um Sistema Distribuído Abertura

- "Um sistema distribuído aberto é um sistema que oferece serviços de acordo com as regras padronizadas que descrevem a sintaxe e a semântica desses serviços".
  - Uso da **IDL**
- Especificações devem ser **completas** e **neutras**.
- Assim sendo importantes para **interoperabilidade** e **portabilidade**.
- O sistema distribuído deve ser **extensível**.

Andrew S. Tanenbaum
Maarten Van Steen

Metas de um Sistema Distribuído Abertura

- Necessário separar política e mecanismo.
  - Muitos sistemas mais antigos e outros contemporâneos são construídos com uma abordagem **monolítica**.
  - Em uma abordagem relativamente nova é crucial que o sistema seja organizado como um conjunto de componentes relativamente pequenos e de fácil distribuição.

Exercício

- Em busca rápida, veja como 2 dos sistemas abaixo tratam a questão da abertura

  - Facebook
  - Twitter
  - Google Maps
  - Waze
  - Netflix
  - Whatsapp
  - Telegram
  - Instagram

IV-Escalabilidade

- Suportar aumento de carga e de clientes conectados
  - Sem perda de performance
  - Sem aumento de custos desnecessários
  - Evitando gargalos (bottlenecks)
- Uso de elasticidade, virtualização e replicação
- Modelos arquiteturais descentralizados ou menos dependentes de um único ponto (gargalo)

Metas de um Sistema Distribuído Escalabilidade

- Medida, no mínimo, quanto ao tamanho
- Em Termos geográficos
- Em Termos administrativos
- Em termos de usuários
- Em termos de subsistemas

Problemas de escalabilidade – Tamanho

Conceito	Exemplo
Serviços centralizados	Um único servidor para todos os usuários
Dados centralizados	Uma única lista telefônica on-line
Algoritmos centralizados	Fazer roteamento com base em informações completas

Tabela 1.2 *Exemplos de limitações de escalabilidade.*

Problemas de escalabilidade

- O que distingue um algoritmo descentralizado de um algoritmo centralizado?
  - Nenhuma máquina tem informações completas sobre o estado do sistema.
  - As máquinas tomam decisões tendo como base somente informações locais.
  - A falha de uma máquina não arruína o algoritmo.
  - Não há nenhuma premissa implícita quanto à existência de um relógio global

Andrew S. Tanenbaum
Maarten Van Steen

Problemas de escabilidade geográfica

- Impossível prover comunicação síncrona para grandes distâncias;
- Comunicação em redes de longa distância não é confiável e ponto a ponto;
- Soluções centralizadas atrapalham a escalabilidade de tamanho.

Problemas de escalabilidade administrativa

- Difícil estabelecer políticas de uso e pagamento de:
  - recursos;
  - gerenciamento; e
  - segurança

Técnicas de Escalabilidade
Comunicação síncrona X assíncrona

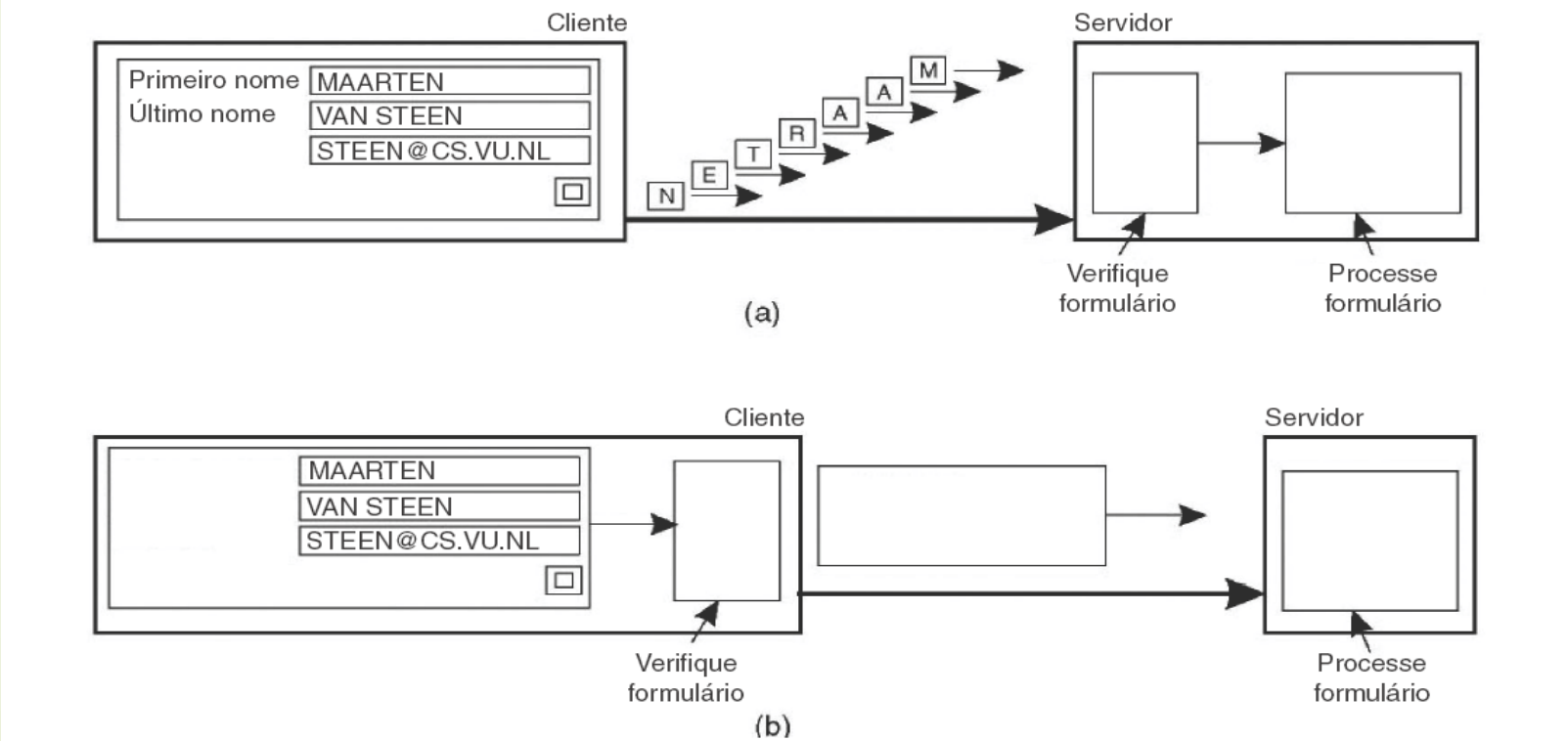


Figura 1.2 *A diferença entre deixar (a) um servidor ou (b) um cliente verificar formulários à medida que são preenchidos.*

Andrew S. Tanenbaum
Maarten Van Steen

Técnicas de escalabilidade - distribuição

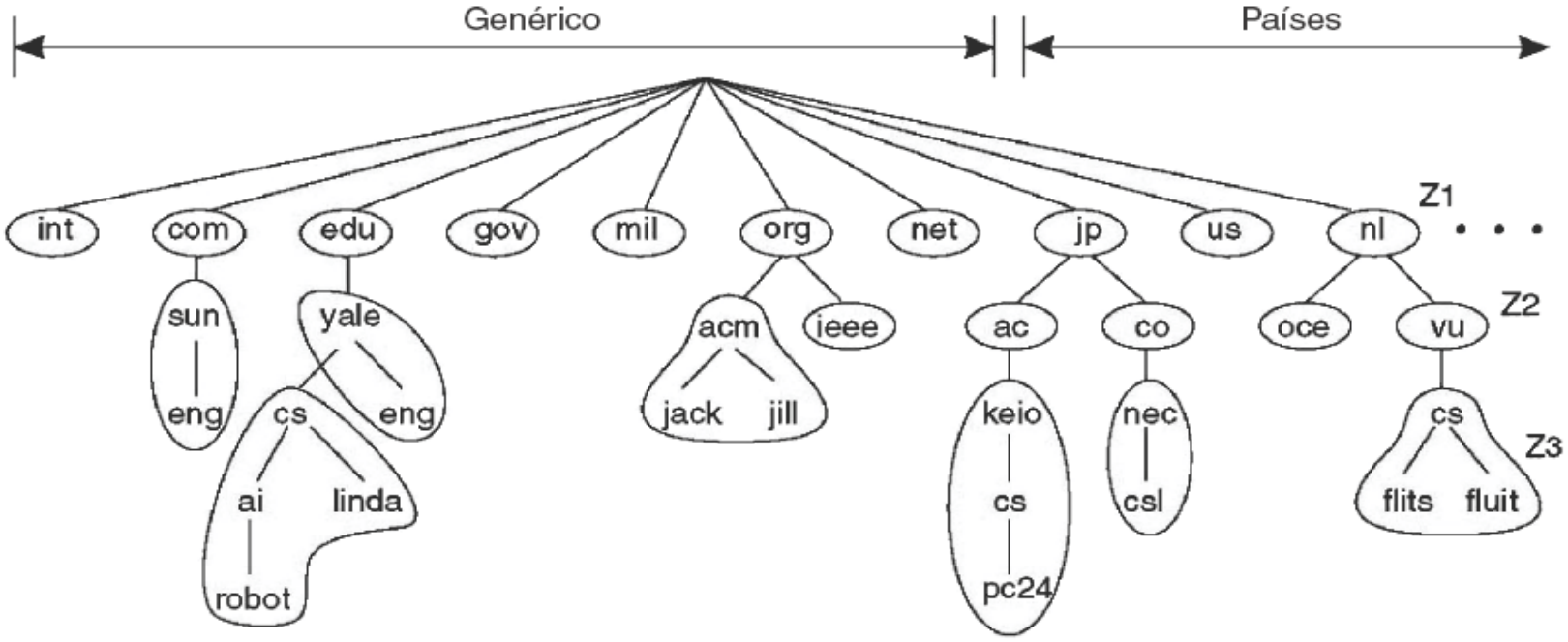
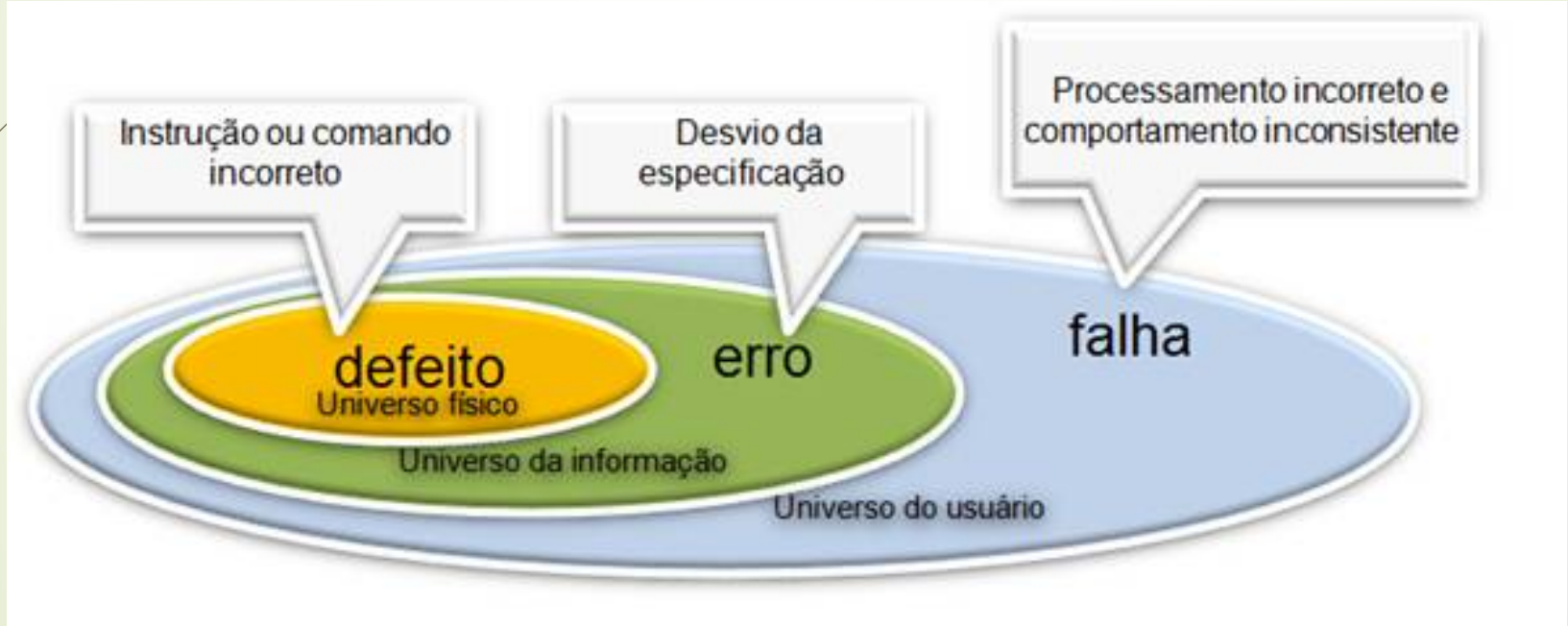


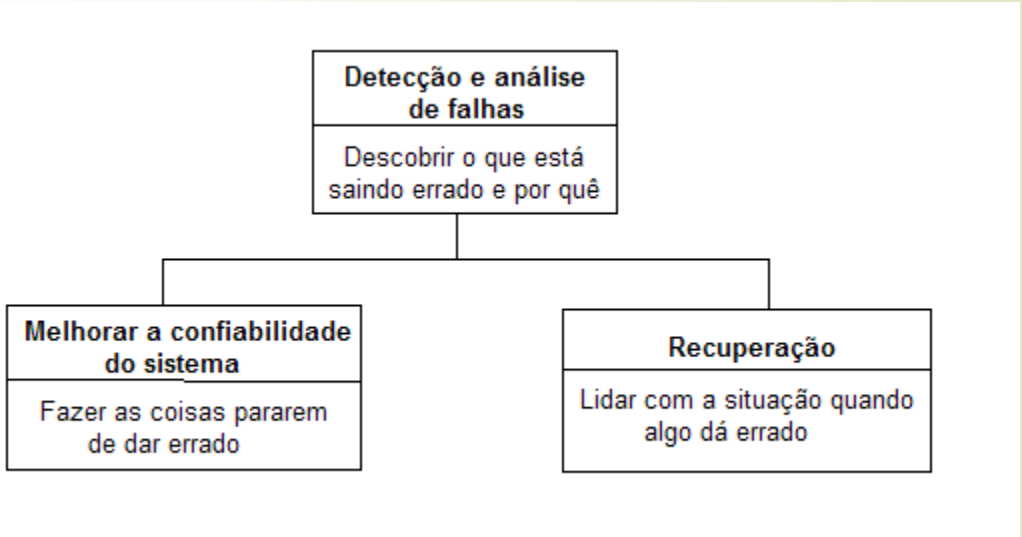
Figura 1.3 Exemplo de divisão do espaço de nomes do DNS em zonas.

V- Tratamento de Falhas

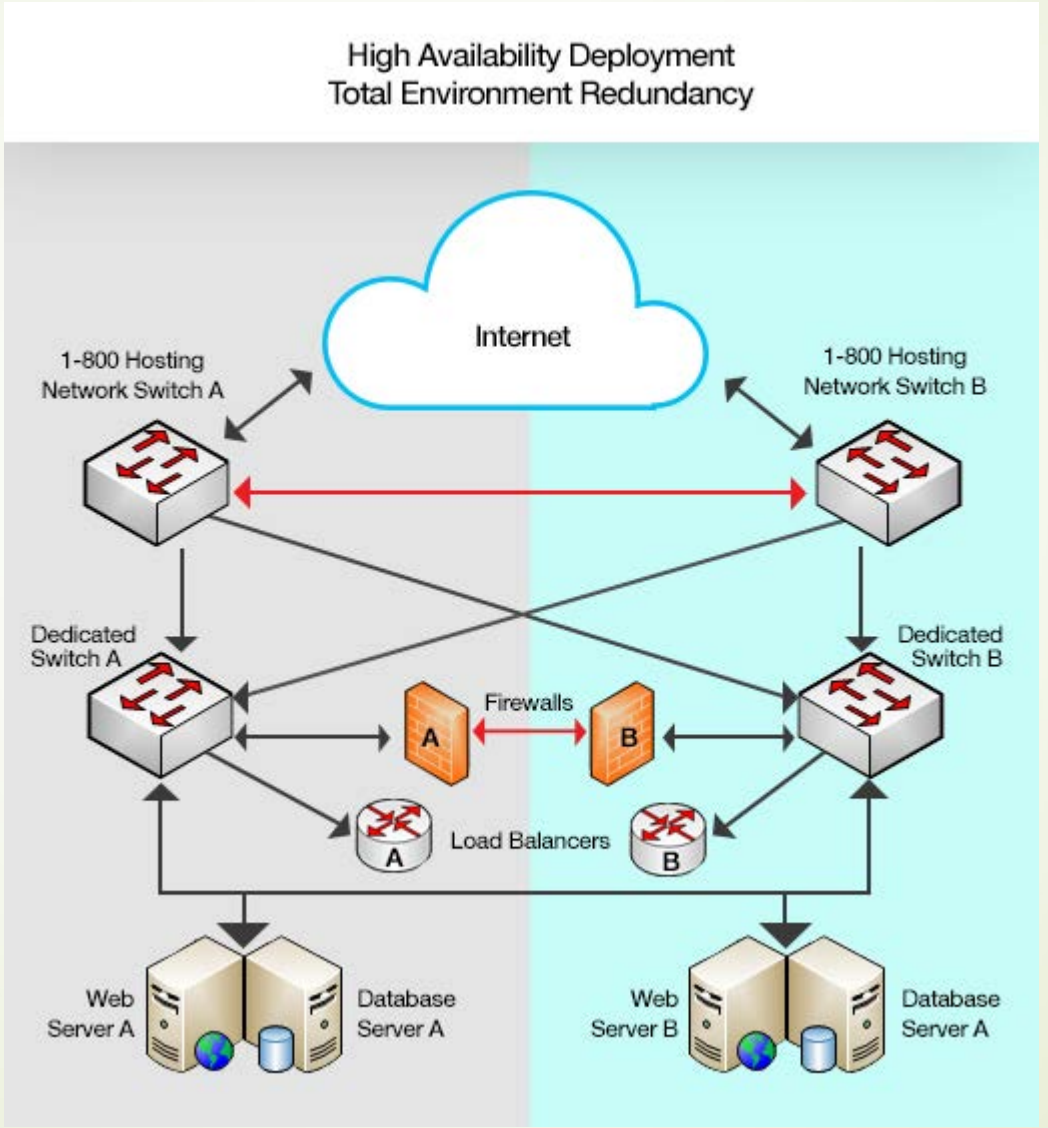


V-Tratamento de Falhas

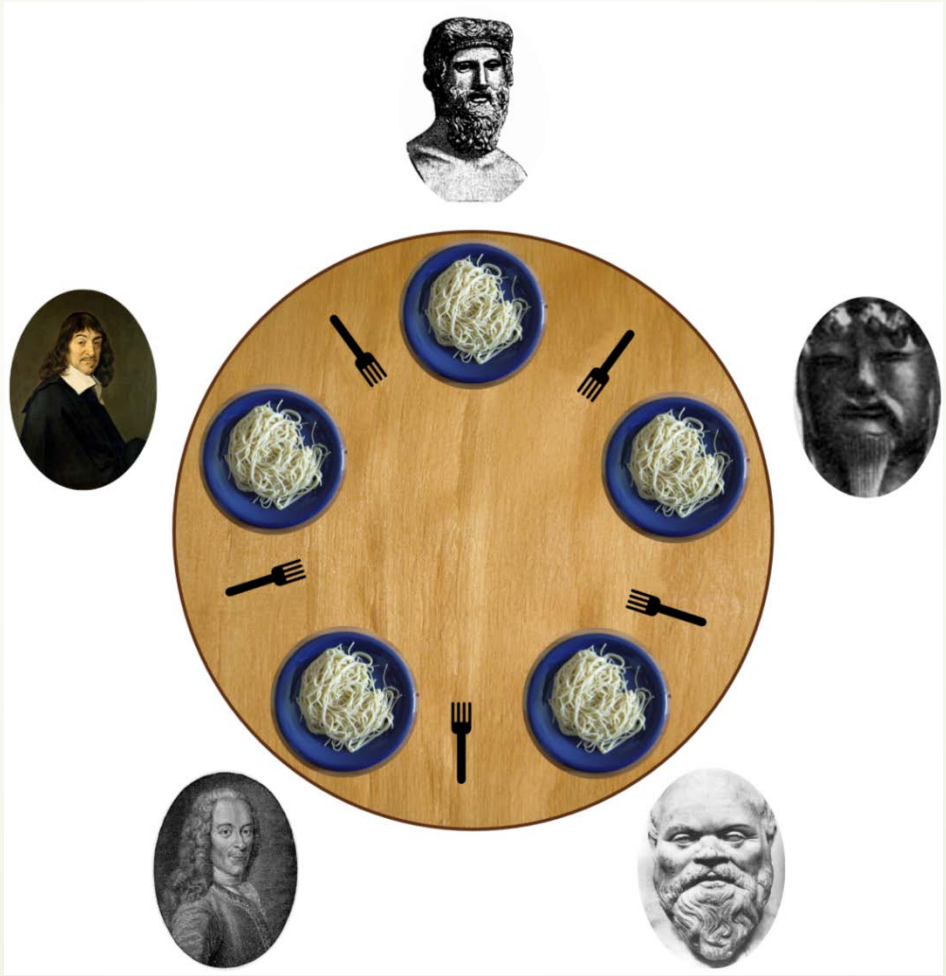
- Detecção de Faltas ou Falhas
- Mascarar Falhas
- Tolerar Falhas
- Recuperação em caso de falhas



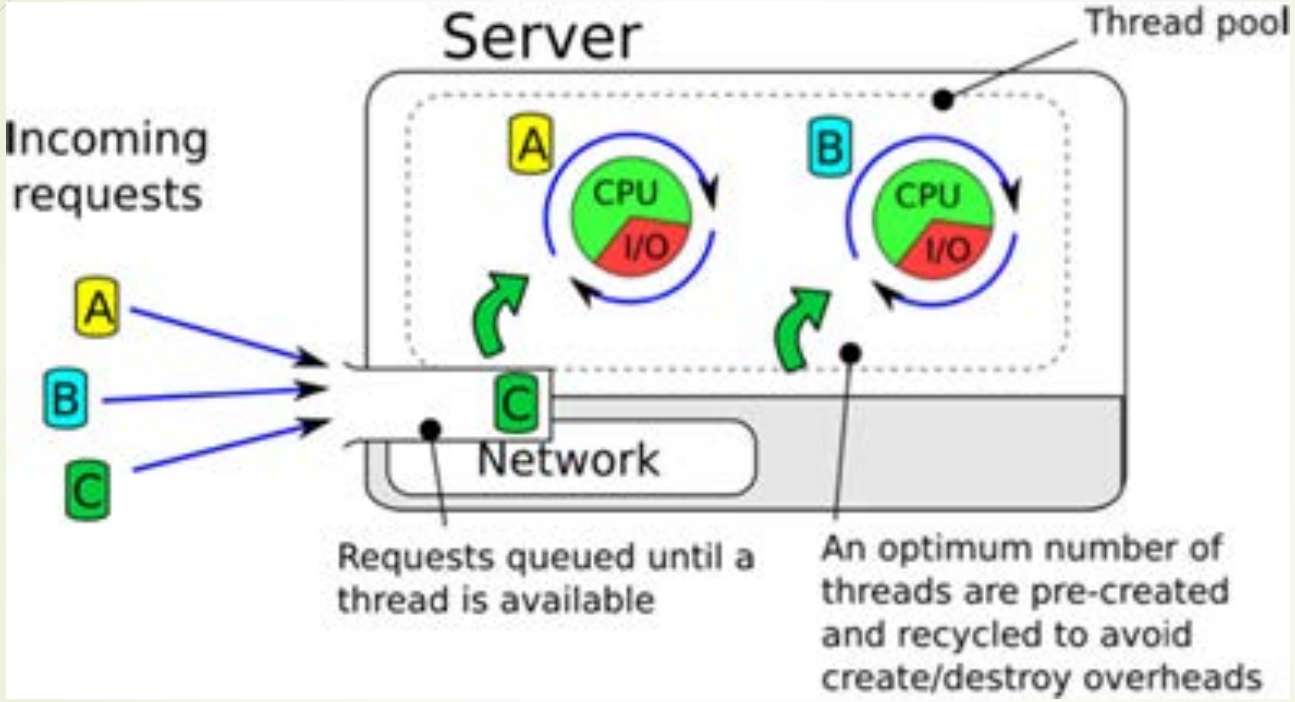
V-Tratamento de Falhas – Soluções?



VI – Concorrência



VI- Concorrência



Controle de Transações

- Empacota várias requisições de programas clientes em uma transação distribuída.
- EAI (Enterprise Application Integration)
- RPCs (Procedimentos remotos)

Primitiva	Descrição
BEGIN_TRANSACTION	Marque o início de uma transação
END_TRANSACTION	Termine a transação e tente comprometê-la
ABORT_TRANSACTION	Elimine a transação e restaure os valores antigos
READ	Leia dados de um arquivo, tabela ou de outra forma
WRITE	Escreva dados para um arquivo, tabela ou de outra forma

Tabela 1.3 Exemplos de primitivas para transações.

Propriedades das transações

1. **A**tômicas: para o mundo exterior, a transação acontece como se fosse indivisível.
2. **C**onsistentes: a transação não viola invariantes de sistema.
3. **I**soladas: transações concorrentes não interferem umas nas outras.
4. **D**uráveis: uma vez comprometida uma transação, as alterações são permanentes

- **ACID** (para facilitar a decoreba)

Andrew S. Tanenbaum
Maarten Van Steen

Exemplo de Transação Aninhada

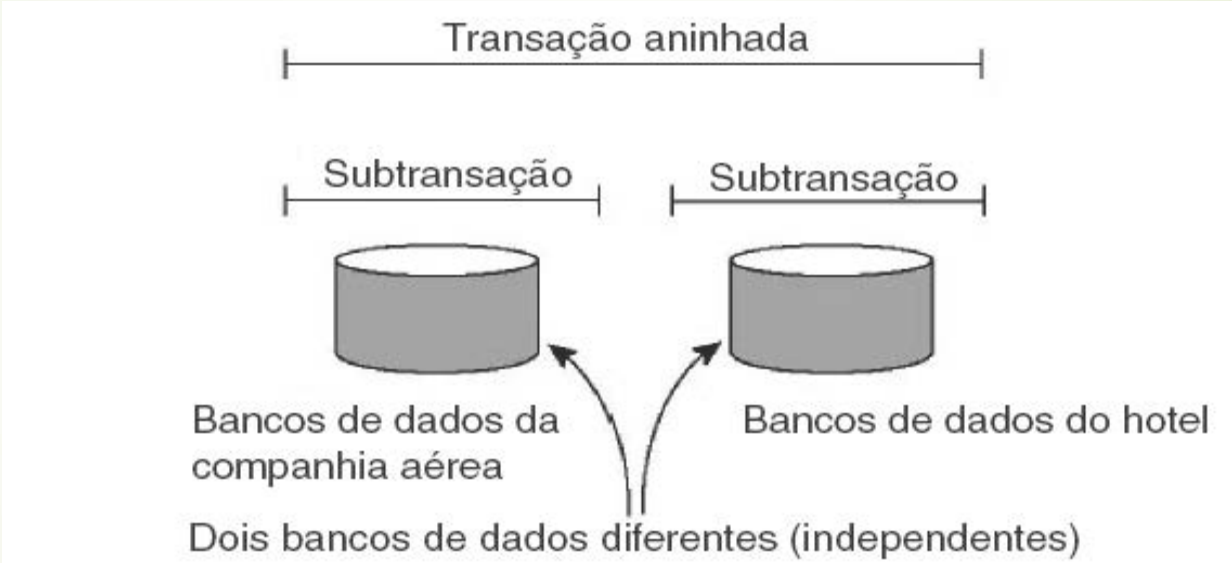


Figura 1.6 Transação aninhada.

Andrew S. Tanenbaum
Maarten Van Steen

Integração usando Monitor TP

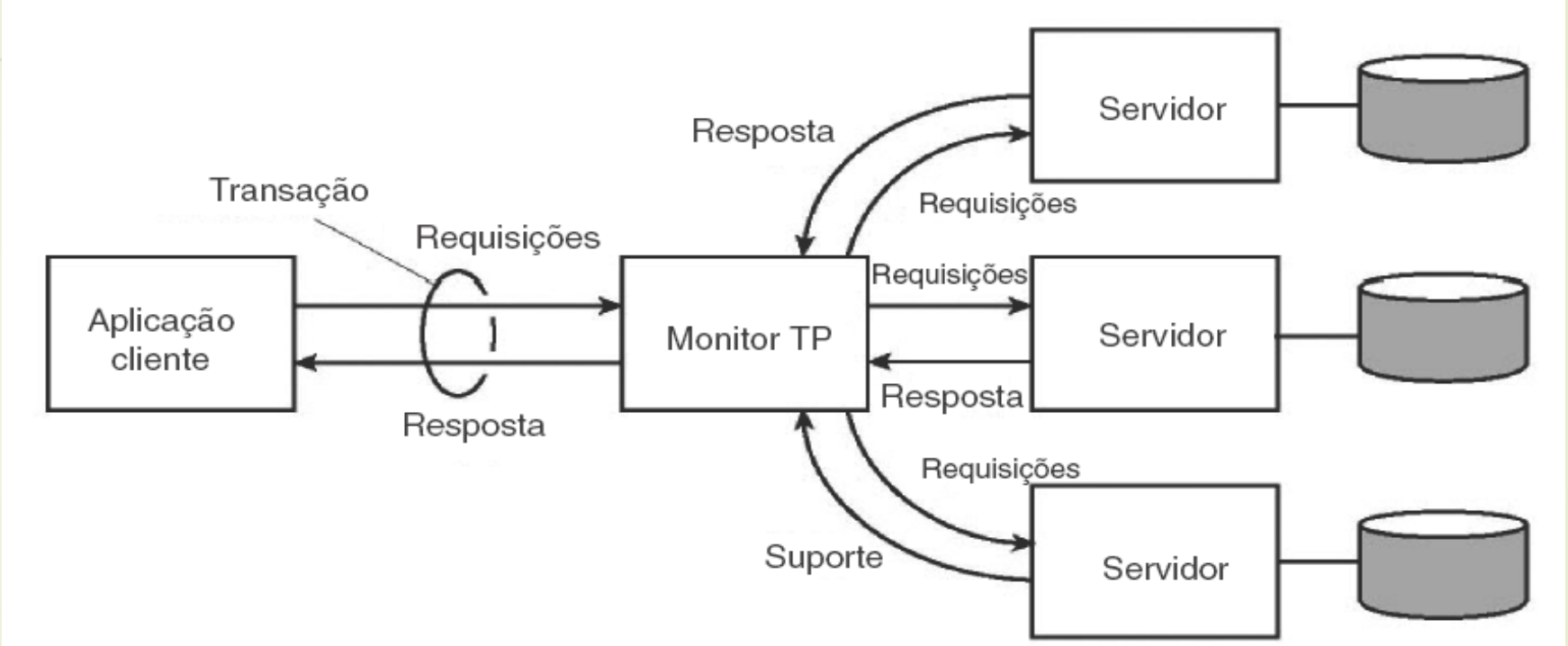


Figura 1.7 O papel do monitor TP em sistemas distribuídos.

Andrew S. Tanenbaum
Maarten Van Steen

VII- Transparência

Metas de um Sistema Distribuído
Transparência da distribuição

Transparência	Descrição
Acesso	Oculta diferenças na representação de dados e no modo de acesso a um recurso
Localização	Oculta o lugar em que um recurso está localizado
Migração	Oculta que um recurso pode ser movido para outra localização
Relocação	Oculta que um recurso pode ser movido para uma outra localização enquanto em uso
Replicação	Oculta que um recurso é replicado
Concorrência	Oculta que um recurso pode ser compartilhado por diversos usuários concorrentes
Falha	Oculta a falha e a recuperação de um recurso

Tabela 1.1 *Diferentes formas de transparência em um sistema distribuído (ISO, 1995).*

Metas de um Sistema Distribuído
Transparência da distribuição

- Grau de transparência – Deve levar em consideração várias questões, como desempenho e facilidade de compreensão.
  - O usuário deve saber das limitações do sistema decorrentes do mesmo ser distribuídos, como:
    - Tempo de acesso?
    - Localização de recursos?

Desenvoler Software Distribuído é simples?

Principais ciladas

- Premissas falsas adotadas ao desenvolver uma aplicação distribuída pela primeira vez
  - A rede é confiável
  - A rede é segura
  - A rede é homogênea
  - A topologia não muda
  - A latência é zero
  - A largura da banda é infinita
  - O custo de transporte é zero
  - Há apenas um administrador

Outros exemplos de Sistemas Distribuídos

Tipos de Sistema Distribuído - Cluster

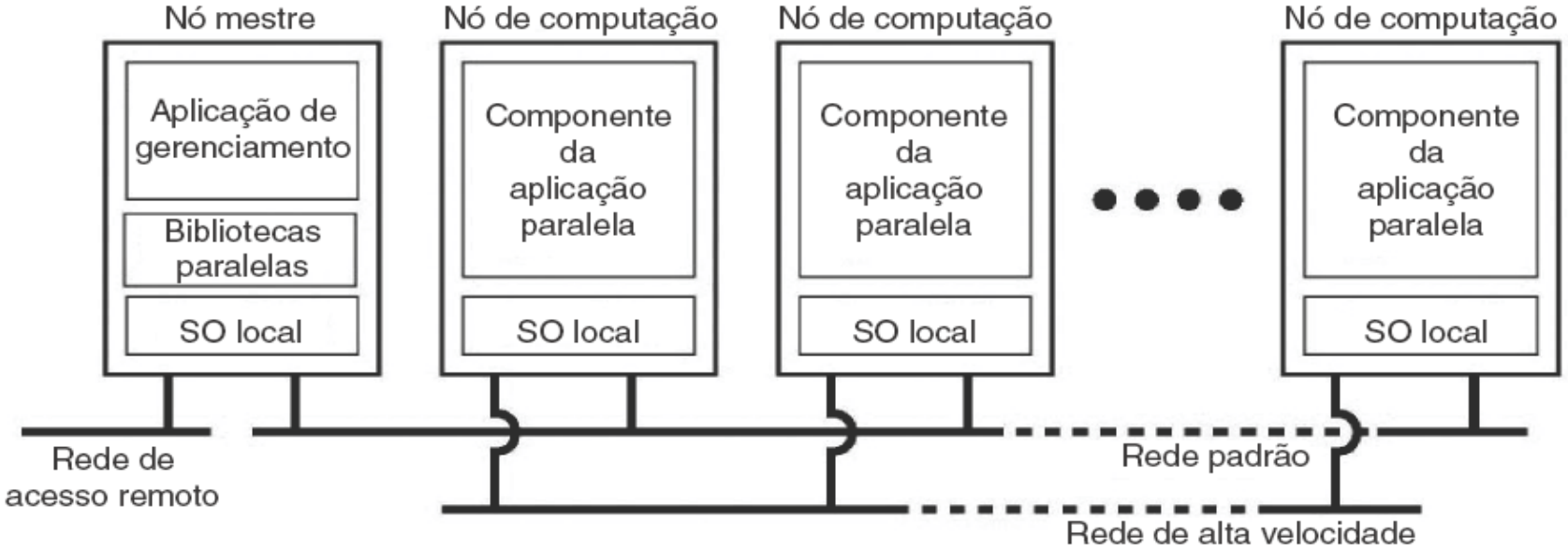


Figura 1.4 Exemplo de um sistema de computação de cluster.

Andrew S. Tanenbaum
Maarten Van Steen

Sistemas Distribuídos Pervasivos

- Sistemas decorrentes do uso de computação móvel e embutida, nas quais o comportamento esperado é a instabilidade;
  - Pequeno tamanho
  - Alimentados por bateria;
  - Comunicação sem fio;
- Não possui controle administrativo humano, podendo:
  1. Adotar mudanças contextuais
  2. Incentivar composição ad hoc
  3. Reconhecer compartilhamento como padrão

Andrew S. Tanenbaum
Maarten Van Steen

Sistemas Pervasivos - Exemplos
Sistemas para tratamento de Saúde

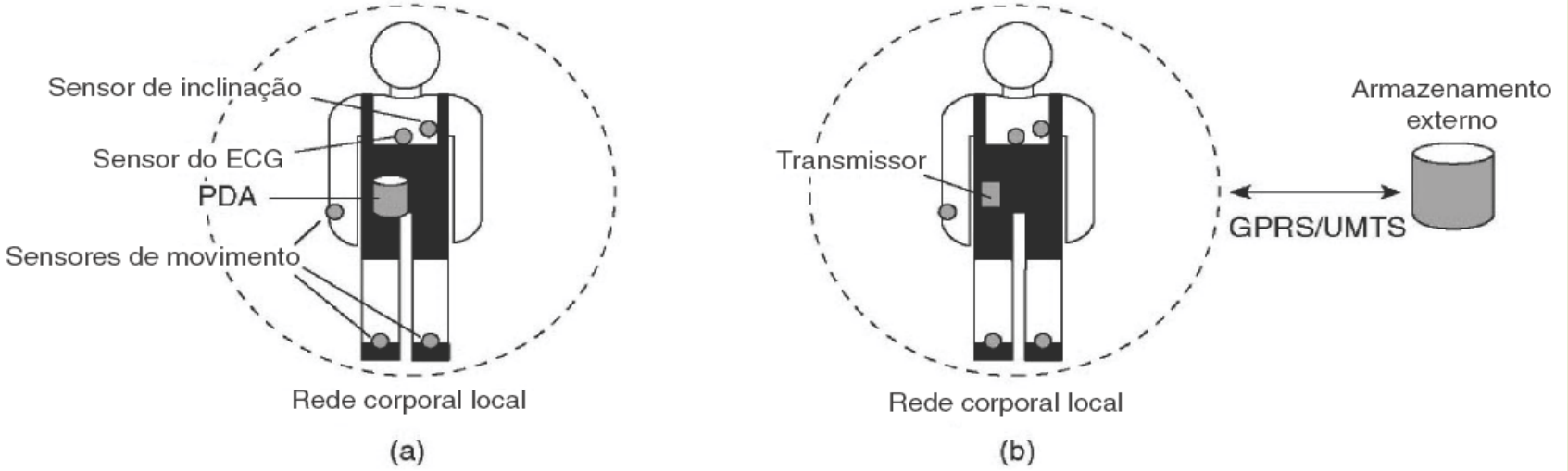
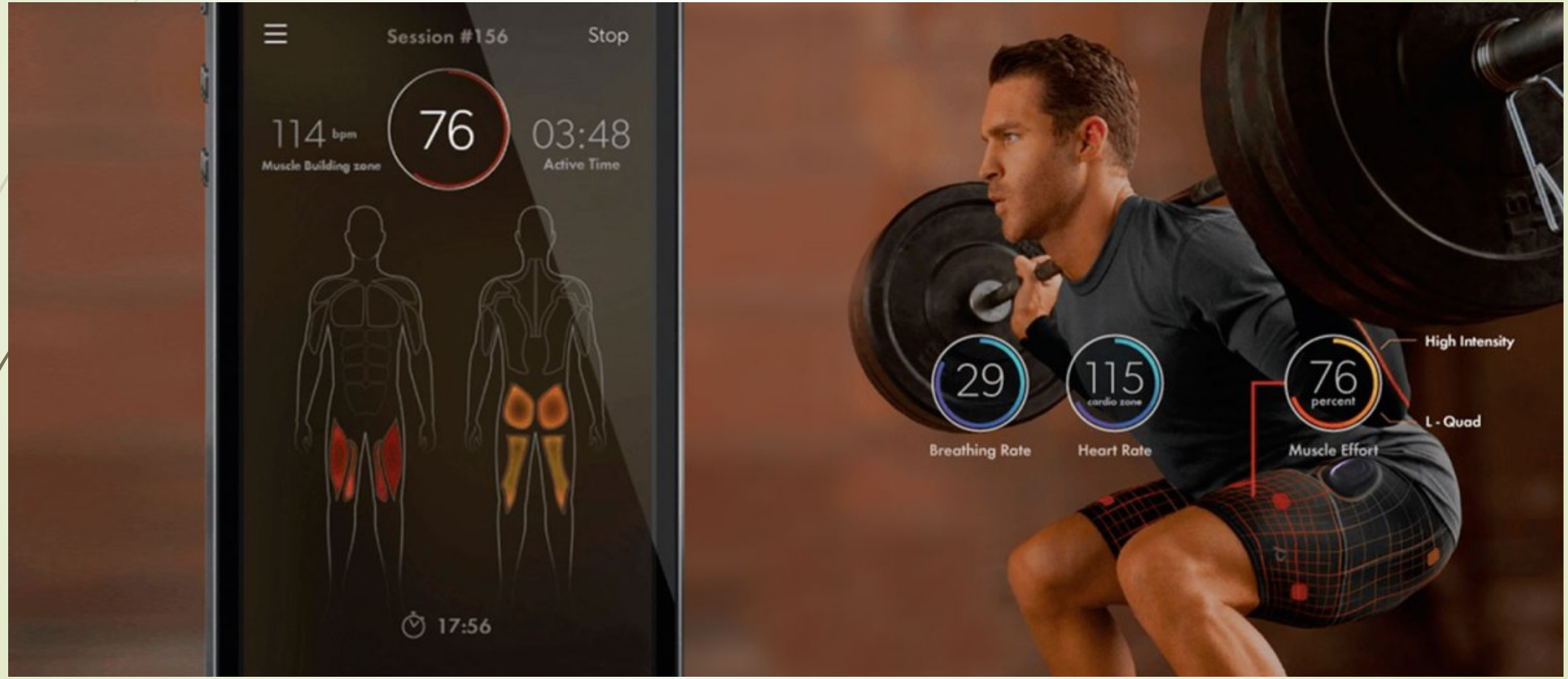


Figura 1.9 Monitoração de uma pessoa em um sistema eletrônico pervasivo de tratamento de saúde utilizando (a) um hub local ou (b) uma conexão contínua sem fio.

Andrew S. Tanenbaum
Maarten Van Steen

Sistemas Pervasivos - Exemplos
Sistemas para Saúde/Fitness



Andrew S. Tanenbaum
Maarten Van Steen

Aula Invertida na próxima semana

- Não esquecer!