



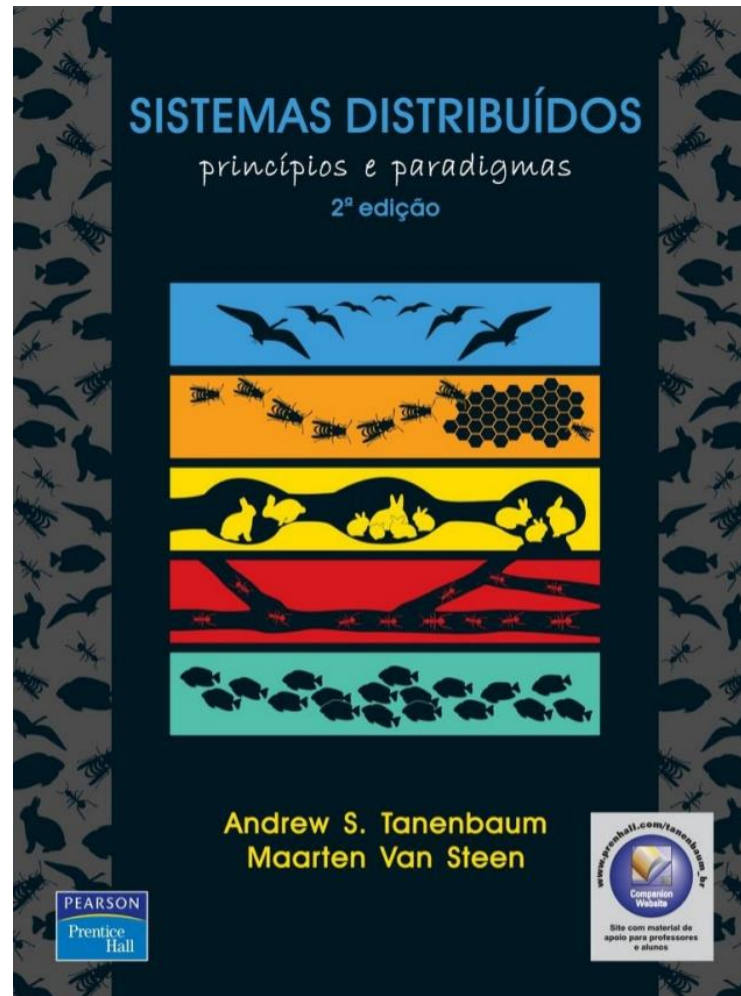
PROVAS DE TI
TUDO PARA VOCÊ PASSAR

Sistemas Distribuídos

http://www.provasdeti.com.br/por-professor/col3/rodrigo-macedo?__SID=U

<http://www.itnerante.com.br/profile/RodrigoMacedo>

Bibliografia



<http://www.provasdeti.com.br/mdl-iso20000-iso-20000-2005-prof-rodriogo-macedo.html>



<http://www.provasdeti.com.br/mdl-tddex-tdd-em-exercicios-prof-r-macedo.html>



Escopo do módulo

Lição 01 - Conceitos e Metas

Lição 02 - Questões

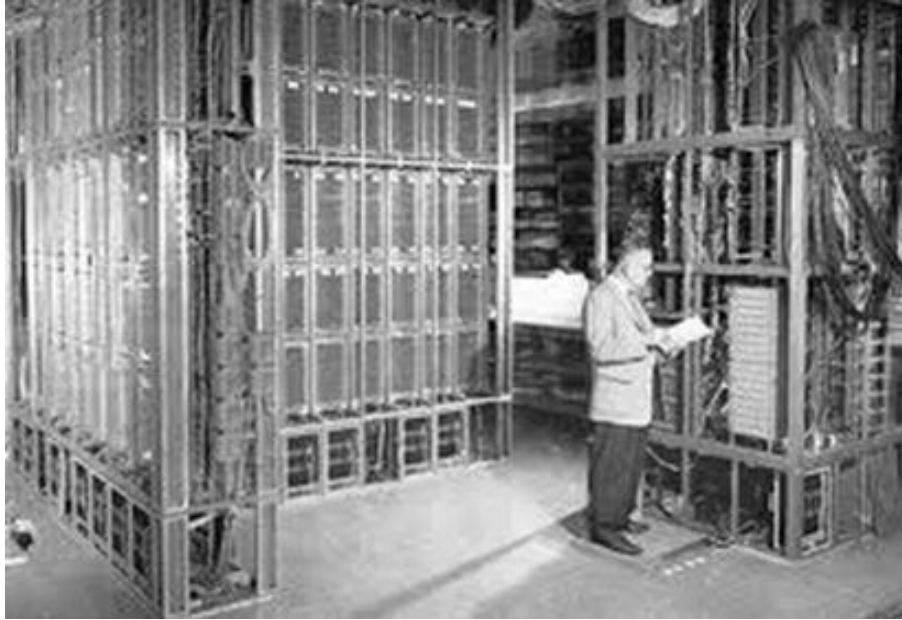
Lição 03 - Modelos Arquiteturais

Lição 04 - Middleware

Lição 05 - Middleware orientado a mensagens

Lição 06 - API e Web Service

Cenário década de 50



Sistemas Distribuídos



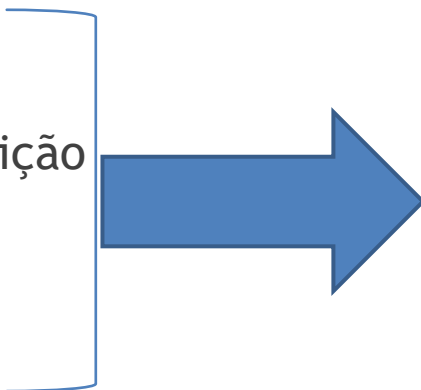
“Um sistema distribuído é formado por um conjunto de computadores **independentes**, que se apresenta para o usuário como sendo um **sistema único** e coerente” - Andrew Tanenbaum.

Vantagens e desvantagens

Vantagens	Desvantagens
Independência de Hardware.	Custo maior de desenvolvimento.
Compartilhamento de recursos.	Maior probabilidade de ocorrência de falhas no sistema.
Maior poder de processamento.	Poucos softwares disponíveis.
Mais disponibilidade	Segurança.
Independência de Sistema Operacional.	Sobrecarga na rede.

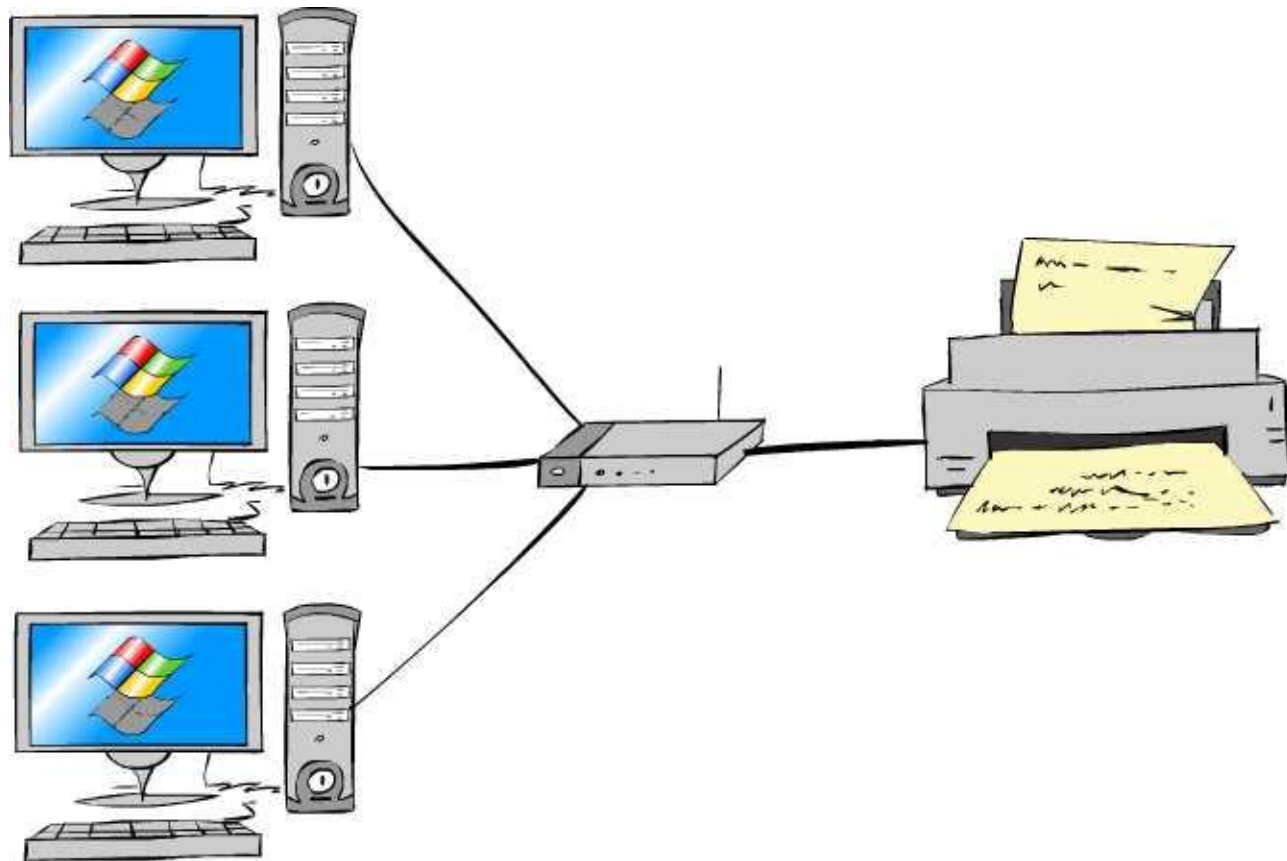
Metas

1. Acesso a recursos
2. Transparência de distribuição
3. Abertura
4. Escalabilidade



Acesso a recursos

- Uma das principais metas de Sistemas Distribuídos.
- Dispositivos ou componentes heterogêneos (Independência hardware e software).



Transparência de Distribuição

- “Um sistema distribuído deve se apresentar como um sistema único”.
- 1. Transparência a falhas.
- 2. Transparência a concorrências.
- 3. Transparência a relocação.
- 4. Transparência de acesso.
- 5. Transparência de localização.
- 6. Transparência de migração.
- 7. Transparência de replicação.



Transparência a falhas

- “Você sabe que tem um sistema distribuído quando uma falha em um computador que você nunca ouviu falar impede de você realizar o seu trabalho” - Leslie Lamport.
- Apesar de haver falhas nos sistemas distribuídos.
- Devem ficar ocultas para o usuário.
- E a melhor forma de implementar isso, é quando uma falha ocorre, e ainda assim não impede que os usuários realizem seu trabalho, ainda que o desempenho venha a ser comprometido.



Transparência a concorrências

- Quando se trata de acesso a recursos, pode-se ter via duas perspectivas: Cooperativo e Competitivo.
- Deve ficar oculto para o usuário que outros usuários podem estar requisitando o mesmo recurso que o seu.
- Ex: Requisição de impressão.



Transparência a relocação

- Recursos podem ser relocados enquanto estão sendo acessados.
- Porém sem que o usuário perceba.
- Ex: Uso móvel de um notebook através de rede wifi.



Transparência de acesso

- Ocultar do usuário como os dados serão apresentados e acessados.
- Ex: Recursos devem estar disponível em diferentes sistemas operacionais



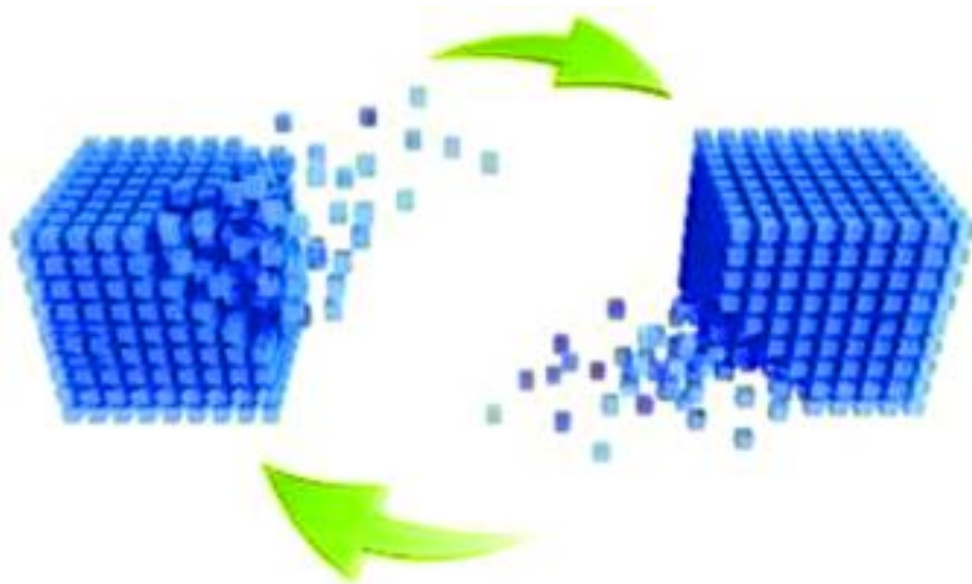
Transparência de localização

- Ocultar do usuário a localização física onde os recursos estão localizados.
- Dar nomes lógicos para abstrair a localização física.
- Ex: URL



Transparência de migração

- Recursos podem ser movimentados sem afetar o modo como são acessados.
- Ex: Migração de um servidor, aplicação, etc.



Transparência de replicação

- Ocultar a existência de duas ou mais cópias de um recurso.
- Ex: Sistemas informatizados a empresas matriz-filial.



Abertura

- Oferece serviços de acordo com a regras padrões (RFC, OMG).
- Outra característica tem a ver com a extensibilidade de um sistema.
- Podem ser ampliados em nível de:
 - Hardware: Adição de computadores a rede.
 - Software: Adição de novos serviços ou atualização dos antigos.
- **Portabilidade:** Característica que permite que um recurso possa estar sendo acessada em diversas arquiteturas e sistemas operacionais diferentes sem necessidade de configuração adicional.
- **Interoperabilidade:** Característica que se define o nível de comunicação e interação que os recursos terão entre si.

Palavra chave: **Padronização.**

Escalabilidade

- Capacidade do crescimento do sistema.
- “É considerado escalável um sistema se este se mantém estável num aumento significativo de recursos e número de usuários”.

Medidas:

- ▶ Crescimento vertical: Investe-se no “poder de fogo”.
 - ▶ A ideia aqui, é criar um **supercomputador** que consiga atender a demanda atual e futura da empresa
- ▶ Crescimento horizontal: Investe-se na compra de muitos computadores.
 - ▶ A ideia aqui é comprar diversos computadores mesmo que com configuração baixa, mas que propicia um ambiente que proporcione um trabalho colaborativo destes. Como é o caso da **clusterização**.

Técnicas de Escalabilidade

1. Distribuição: É interessante às vezes subdividir um recurso em partes menores. Um exemplo desta técnica é o **DNS** que subdivide o domínio em zonas (partes menores).
2. Ocultar a latência de comunicação: A ideia é diminuir o tempo de espera em requisições remotas.
 1. Recomenda-se o uso de comunicação assíncrona ao invés de síncrona.
 2. E que validação de dados possa ser feita no lado do cliente para não sobrecarregar o servidor.
3. Replicação: A replicação é usada em sistemas distribuídos como medida de desempenho e segurança do armazenamento de dados. Um exemplo é o **cache**.

Q2 - [FUNIVERSA CEB-DISTRIBUIÇÃO S/A 2010] Entre os princípios básicos de Sistemas Distribuídos está o da "Transparência", que define formas para uso de computadores independentes e interligados, que se apresentam ao usuário como um sistema único e consistente. Um tipo de transparência é a "Transparência de Acesso". Com relação à definição de transparência de acesso, assinale a alternativa correta.

- a) O acesso ao recurso é transparente, estando este em disco ou na memória.
- b) Uso de múltiplas cópias de um recurso para aumento do desempenho do sistema.
- c) A execução de um processo não interfere na do outro, mesmo com uso compartilhado de recursos.
- d) Padronização das operações de acesso, tanto para recursos locais quanto para remotos.
- e) O acesso ao recurso é feito sem a necessidade de se determinar a localização de tal recurso.

Q2 - [FUNIVERSA CEB-DISTRIBUIÇÃO S/A 2010] Entre os princípios básicos de Sistemas Distribuídos está o da "Transparência", que define formas para uso de computadores independentes e interligados, que se apresentam ao usuário como um sistema único e consistente. Um tipo de transparência é a "Transparência de Acesso". Com relação à definição de transparência de acesso, assinale a alternativa correta.

- a) O acesso ao recurso é transparente, estando este em disco ou na memória.
- b) Uso de múltiplas cópias de um recurso para aumento do desempenho do sistema.
- c) A execução de um processo não interfere na do outro, mesmo com uso compartilhado de recursos.
- d) Padronização das operações de acesso, tanto para recursos locais quanto para remotos.
- e) O acesso ao recurso é feito sem a necessidade de se determinar a localização de tal recurso.

Q3) [2014 FUNPRESP] O conceito de transparência pode ser aplicado a diversos aspectos de um sistema distribuído. Assinale a alternativa correta quanto ao tipo de transparência e a respectiva descrição.

- (A) Acesso mostra o lugar onde um recurso está localizado.
- (B) Replicação oculta que um recurso é replicado.
- (C) Relocação oculta que um recurso não pode ser movido para outra localização estando fora de uso.
- (D) Migração mostra que um recurso pode ser movido para outra localização.
- (E) Falha mostra a falha e recuperação de um recurso.

Q3) [2014 FUNPRESP] O conceito de transparência pode ser aplicado a diversos aspectos de um sistema distribuído. Assinale a alternativa correta quanto ao tipo de transparência e a respectiva descrição.

- (A) Acesso mostra o lugar onde um recurso está localizado.
- (B) Replicação oculta que um recurso é replicado.**
- (C) Relocação oculta que um recurso não pode ser movido para outra localização estando fora de uso.
- (D) Migração mostra que um recurso pode ser movido para outra localização.
- (E) Falha mostra a falha e recuperação de um recurso.

Q4 - [IFMA FSADU 2015] Uma das metas de um sistema distribuído é ocultar o fato de que seus processos e recursos estão fisicamente distribuídos por vários computadores. A capacidade que um sistema tem de movimentar seus recursos sem afetar o modo como podem ser acessados é chamada de transparência de

- a) localização
- b) realocação
- c) replicação
- d) migração
- e) acesso

Q4 - [IFMA FSADU 2015] Uma das metas de um sistema distribuído é ocultar o fato de que seus processos e recursos estão fisicamente distribuídos por vários computadores. A capacidade que um sistema tem de movimentar seus recursos sem afetar o modo como podem ser acessados é chamada de transparência de

- a) localização
- b) realocação
- c) replicação
- d) migração
- e) acesso

Q5 - [UERJ 2015] Escalabilidade é um dos principais objetivos para os projetistas e desenvolvedores de sistemas distribuídos. As três técnicas básicas para aumentar a escalabilidade dos sistemas são:

- A) esconder a latência; distribuição; replicação
- B) centralização; replicação; comunicação síncrona
- C) transparência; esconder a latência; centralização
- D) comunicação síncrona; distribuição; transparência

Q6 - [CESPE ANATEL 2009] Uma das vantagens da arquitetura distribuída é o compartilhamento de recursos, que permite que sistemas, aplicativos e dispositivos periféricos, como discos, impressoras, arquivos, estejam associados a diferentes computadores em uma rede. Uma segunda vantagem é a concorrência, uma vez que vários processos podem operar ao mesmo tempo em diferentes computadores na rede. E, por fim, uma terceira vantagem é a proteção, pois o acesso é feito de forma centralizada.

Q5 - [UERJ 2015] Escalabilidade é um dos principais objetivos para os projetistas e desenvolvedores de sistemas distribuídos. As três técnicas básicas para aumentar a escalabilidade dos sistemas são:

- A) esconder a latência; distribuição; replicação
- B) centralização; replicação; comunicação síncrona
- C) transparência; esconder a latência; centralização
- D) comunicação síncrona; distribuição; transparência

Q6 - [CESPE ANATEL 2009] Uma das vantagens da arquitetura distribuída é o compartilhamento de recursos, que permite que sistemas, aplicativos e dispositivos periféricos, como discos, impressoras, arquivos, estejam associados a diferentes computadores em uma rede. Uma segunda vantagem é a concorrência, uma vez que vários processos podem operar ao mesmo tempo em diferentes computadores na rede. E, por fim, uma terceira vantagem é a proteção, pois o acesso é feito de forma centralizada. **ERRADO.**

Q7 - [CESPE TSE 2007 ADAP] Em um sistema distribuído, os usuários podem acessar recursos em outras máquinas, mas um sistema operacional distribuído não tem como tornar a distribuição dos recursos transparente aos usuários.

Q7 - [CESPE TSE 2007 ADAP] Em um sistema distribuído, os usuários podem acessar recursos em outras máquinas, mas um sistema operacional distribuído não tem como tornar a distribuição dos recursos transparente aos usuários.

ERRADO.

Gabarito

Q1 - LETRA B.

Q2 - LETRA D.

Q3 - LETRA B.

Q4 - LETRA D.

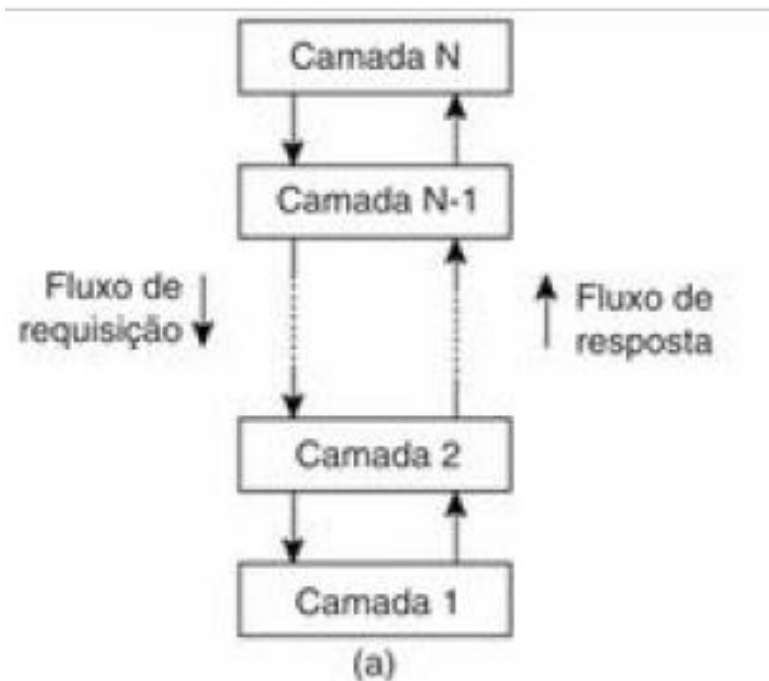
Q5 - LETRA A.

Q6 - ERRADO.

Q7 - ERRADO.

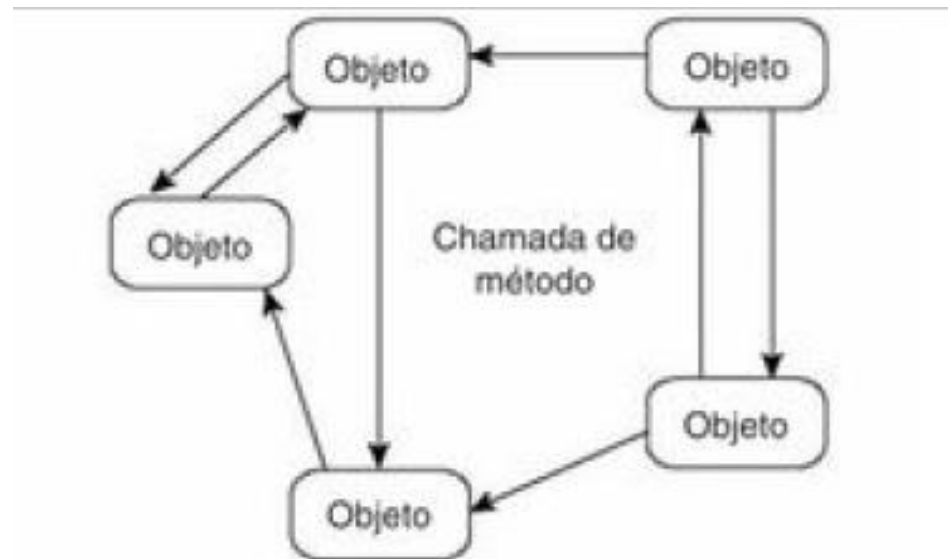
Arquitetura em camadas

- ▶ Um componente tem permissão de chamar outro que esteja imediatamente inferior a ele, e não o inverso.
- ▶ Controle de requisição segue de um nível hierárquico maior para menor, e controle de resposta, o inverso.



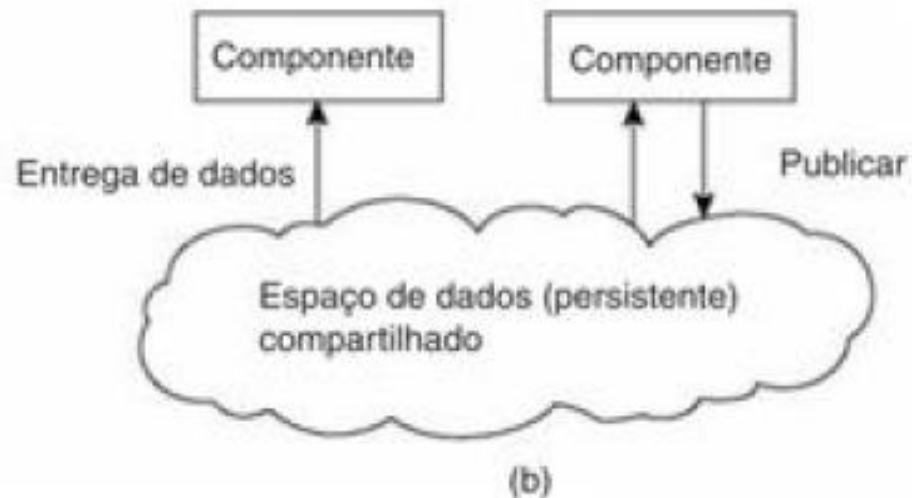
Arquitetura baseada em objetos

- ▶ Cada objeto corresponde a um componente, e esses objetos são interconectados por meio de uma chamada de procedimento (remota).
- ▶ Arquitetura em camadas + Orientada a objetos = formam estilos mais importantes para sistemas de grande porte.



Arquitetura centrada em dados

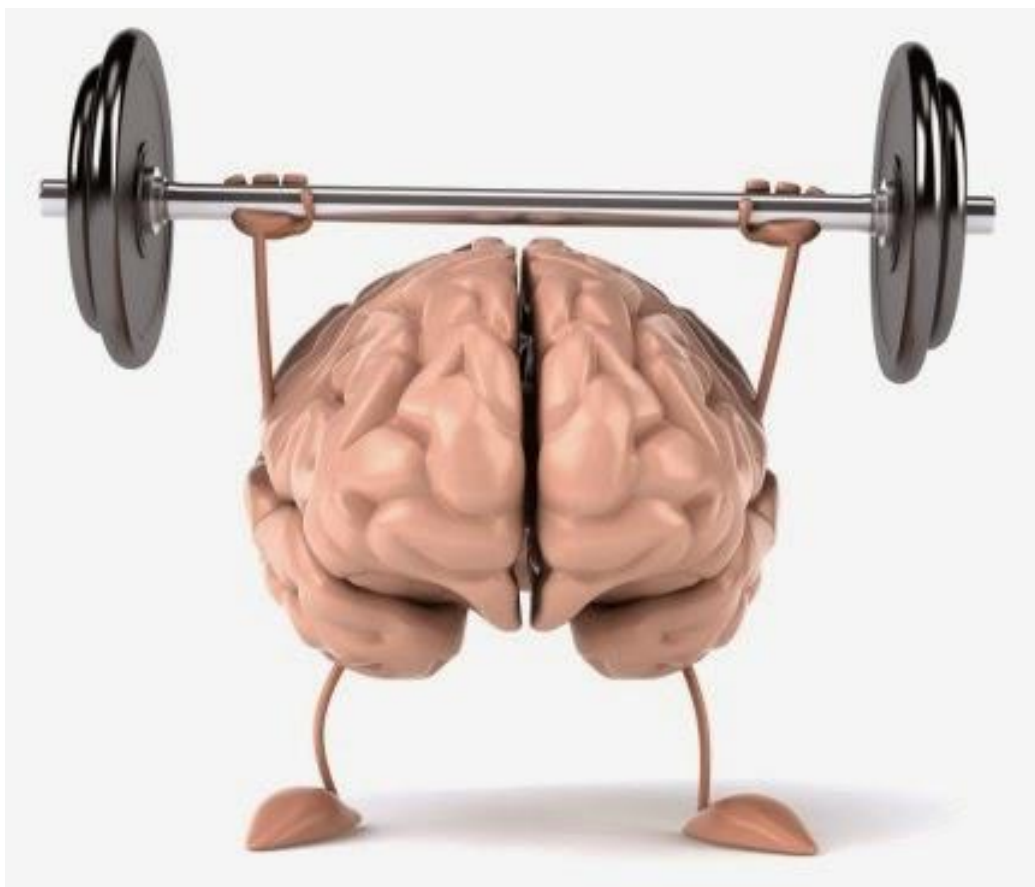
- ▶ Processos se comunicam por meio de um repositório comum (passivo ou ativo).
- ▶ Os componentes podem requisitar ou publicar dados neste repositório.
- ▶ Ex: Dropbox.



Arquitetura baseada em eventos

- ▶ Os processos se comunicam, em essência, por meio de propagação de eventos, opcionalmente, também transportam dados
- ▶ Sistemas “publicar/subscrever”.
- ▶ Processos **publicam** eventos onde o middleware assegura que somente os processos que subscreveram nesse evento o receberão.
- ▶ Tem a presença do middleware que promove o **baixo acoplamento**.





Q1 - [CESPE SERPRO 2013] Na arquitetura distribuída, os sistemas orientados a eventos possuem processos fortemente acoplados.

Q2 - [CESPE SERPRO 2013] Na arquitetura de sistemas distribuídos, a propagação de eventos associa-se com o que se denomina sistemas publicar/subscrever.

Q1 - [CESPE SERPRO 2013] Na arquitetura distribuída, os sistemas orientados a eventos possuem processos fortemente acoplados.
ERRADO.

Q2 - [CESPE SERPRO 2013] Nesse tipo de arquitetura, a propagação de eventos associa-se com o que se denomina sistemas publicar/assubcrever. CERTO.

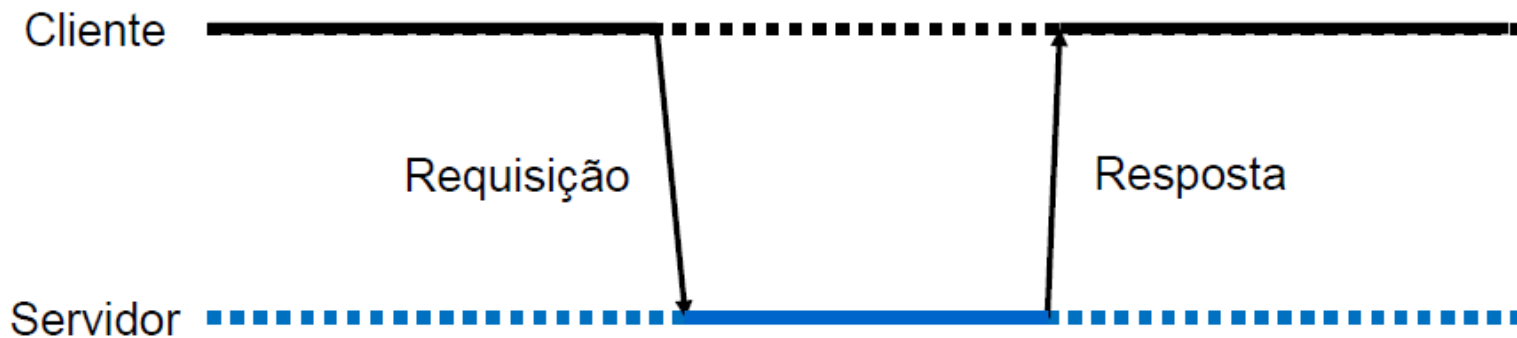
Gabarito

Q1 - ERRADO.

Q2 - CERTO.

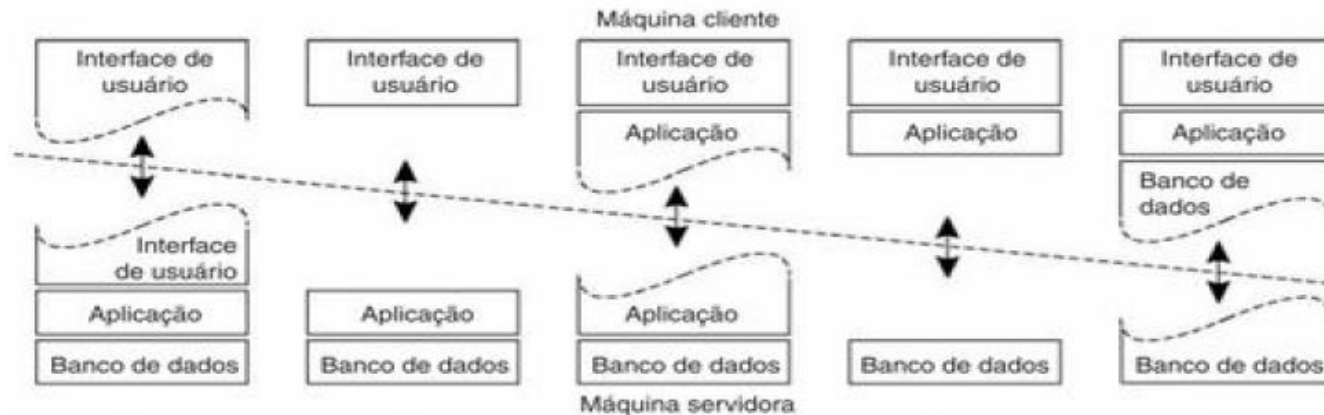
Tipos Arquiteturais (Centralizada)

- ▶ Arquitetura centralizada: famoso cliente-servidor (request - response).
 - ▶ Toda a sobrecarga fica no lado do servidor.



- ▶ Arquitetura MVC: (Model-View-Controller): Há uma separação bem clara das responsabilidades dentro da orquestração de uma aplicação.
 - ▶ **Model:** Aonde se concentra o acesso a dados.
 - ▶ **View:** Nível de interface com o usuário.
 - ▶ **Controller:** Regras de negócio da aplicação.

- ▶ Arquitetura duas camadas: Semelhante ao cliente-servidor. A proporcionalidade de sobrecarga depende da aplicação e pode estar também no cliente.
 - ▶ **Cliente gordo:** Quando o cliente possui muitas responsabilidades na aplicação.
 - ▶ **Cliente magro:** Quando o cliente não possui tantas responsabilidades na aplicação.

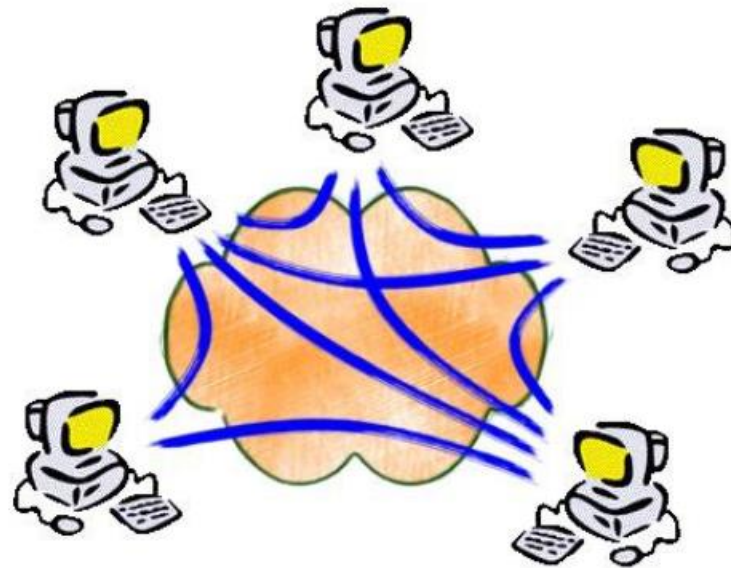


- ▶ Arquitetura de três camadas: Possui responsabilidades bem definida entre suas camadas.



Tipos Arquiteturais (Descentralizada)

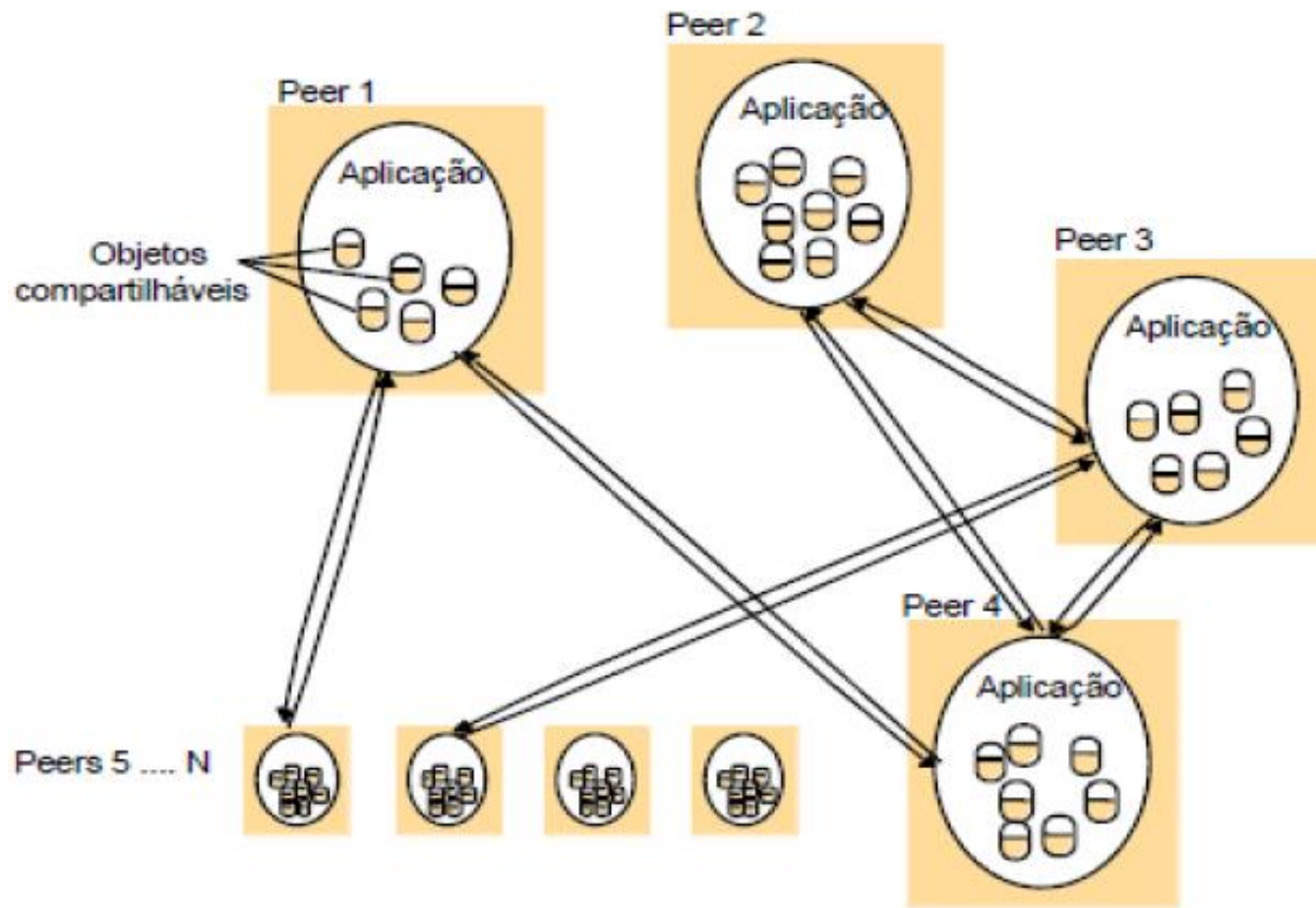
- ▶ Interação é feita de forma horizontal.
- ▶ Cada componente pode se comportar como cliente e servidor ao mesmo tempo.
- ▶ Ex: P2P (Não necessitam de servidor central).



Peer To Peer - Arquitetura Descentralizada

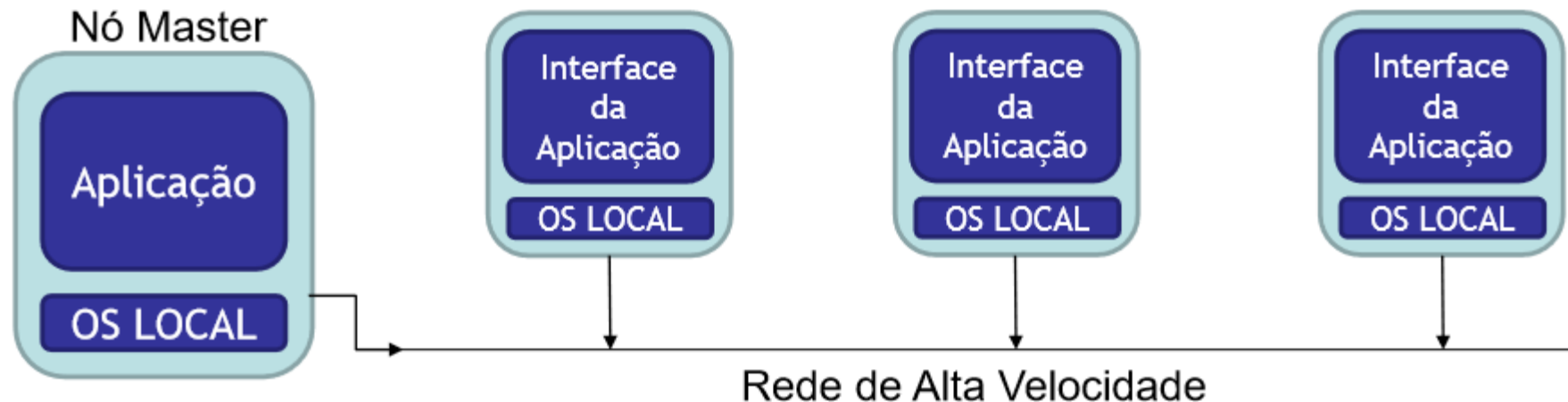
- ▶ Interação é feita de forma horizontal.
- ▶ Cada componente pode se comportar como cliente e servidor ao mesmo tempo.
- ▶ Ex: P2P (Não necessitam de servidor central).
- ▶ Maior escalabilidade comparado com o cliente / servidor.

Exemplo P2P



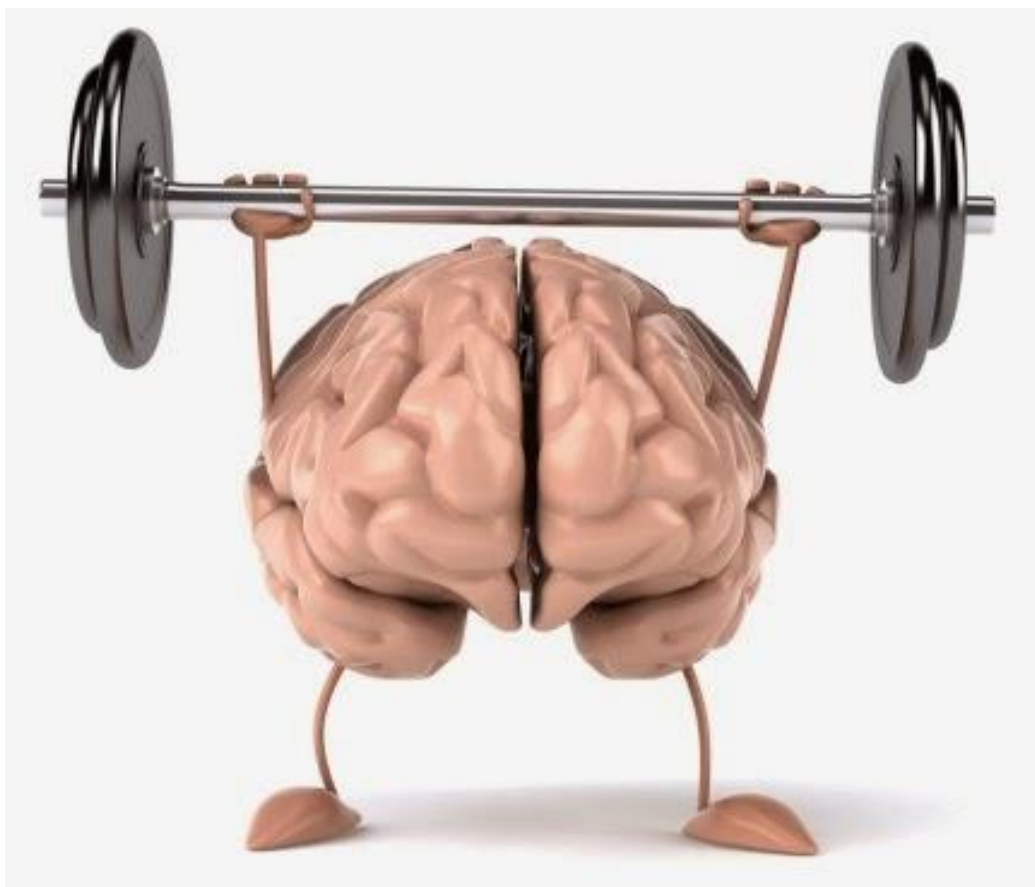
Cluster

- Utilização de máquinas/estações simples com basicamente a mesma estrutura e sistema operacional, interconectadas em uma rede interna de alta velocidade utilizada, geralmente, para processamento paralelo de uma funcionalidade/aplicação



GRID

- ▶ Utilização de máquinas/estações simples, interconectadas em uma rede de alta velocidade. A diferença para sistemas em cluster é que geralmente, as máquinas utilizadas são distintas, com sistemas operacionais diferentes e estruturas diferentes (**heterogêneo**).
- ▶ Além disso, permite o uso compartilhado de recursos computacionais de alta capacidade **distribuídos geograficamente**.



Q1 - [IBFC 2016 EBSERH] Conceito técnico que está diretamente relacionado aos sistemas de alta disponibilidade e que permite agregar vários computadores, ou servidores, como se fossem uma única máquina de grande porte:

- a)cluster
- b)drove
- c)flock
- d)bunch
- e)crowd

Q2 -[CESPE 2014 TJ SE GAB C] Diferentemente de um cluster, uma grade computacional (grid) permite o uso compartilhado de recursos computacionais de alta capacidade distribuídos geograficamente.

Q1 - [IBFC 2016 EBSERH] Conceito técnico que está diretamente relacionado aos sistemas de alta disponibilidade e que permite agregar vários computadores, ou servidores, como se fossem uma única máquina de grande porte:

a)cluster

b)drove

c)flock

d)bunch

e)crowd

Q2 -[CESPE 2014 TJ SE GAB C] Diferentemente de um cluster, uma grade computacional (grid) permite o uso compartilhado de recursos computacionais de alta capacidade distribuídos geograficamente. CERTO.

Q3 - [CESPE 2016 PC PE] Acerca das características dos protocolos de redes peer-to-peer, assinale a opção correta.

- a) É impossível haver combinação com elementos cliente/servidor em aplicações de arquiteturas híbridas.
- b) Os pares de hospedeiros são propriedade dos provedores que prestam serviços aos usuários finais.
- c) Um grande conjunto de servidores é necessário para atender requisições dos demais pares.
- d) Os servidores responsáveis por realizar as trocas de dados entre os pares devem ser dedicados.
- e) Os pares executam aplicações capazes de solicitar e prestar os mesmos serviços, de maneira escalável

Q3 - [CESPE 2016 PC PE] Acerca das características dos protocolos de redes peer-to-peer, assinale a opção correta.

- a) É impossível haver combinação com elementos cliente/servidor em aplicações de arquiteturas híbridas.
- b) Os pares de hospedeiros são propriedade dos provedores que prestam serviços aos usuários finais.
- c) Um grande conjunto de servidores é necessário para atender requisições dos demais pares.
- d) Os servidores responsáveis por realizar as trocas de dados entre os pares devem ser dedicados.
- e) Os pares executam aplicações capazes de solicitar e prestar os mesmos serviços, de maneira escalável

Q4 - [FGV SEPOG RO 2017] Uma rede de computadores em que todos os participantes compartilham seus recursos com todos os outros computadores, sem o uso de computadores centralizadores, é conhecida como rede

a)ad hoc.

b)cliente/servidor.

c)hierárquica.

d)peer-to-peer.

e)publisher/subscriber.

Q5 - [UFBA 2014 UFOB] Uma alternativa para conseguir melhorar o desempenho de alguns sistemas computacionais é a adoção de computação em aglomerados (cluster) e de computação em grades (grids), em que, geralmente, os aglomerados utilizam hardwares, sistemas operacionais e softwares heterogêneos, e as grades, hardwares similares e o mesmo sistema operacional.

Q4 - [FGV SEPOG RO 2017] Uma rede de computadores em que todos os participantes compartilham seus recursos com todos os outros computadores, sem o uso de computadores centralizadores, é conhecida como rede

a)ad hoc.

b)cliente/servidor.

c)hierárquica.

d)peer-to-peer.

e)publisher/subscriber.

Q5 - [UFBA 2014 UFOB] Uma alternativa para conseguir melhorar o desempenho de alguns sistemas computacionais é a adoção de computação em aglomerados (cluster) e de computação em grades (grids), em que, geralmente, os aglomerados utilizam hardwares, sistemas operacionais e softwares heterogêneos, e as grades, hardwares similares e o mesmo sistema operacional. **ERRADO.**

Sistemas Distribuídos Middlewares

Prof. Rodrigo Macedo

Integração de aplicações

Interessantes aplicações **raramente** funcionam de maneira **isolada**.

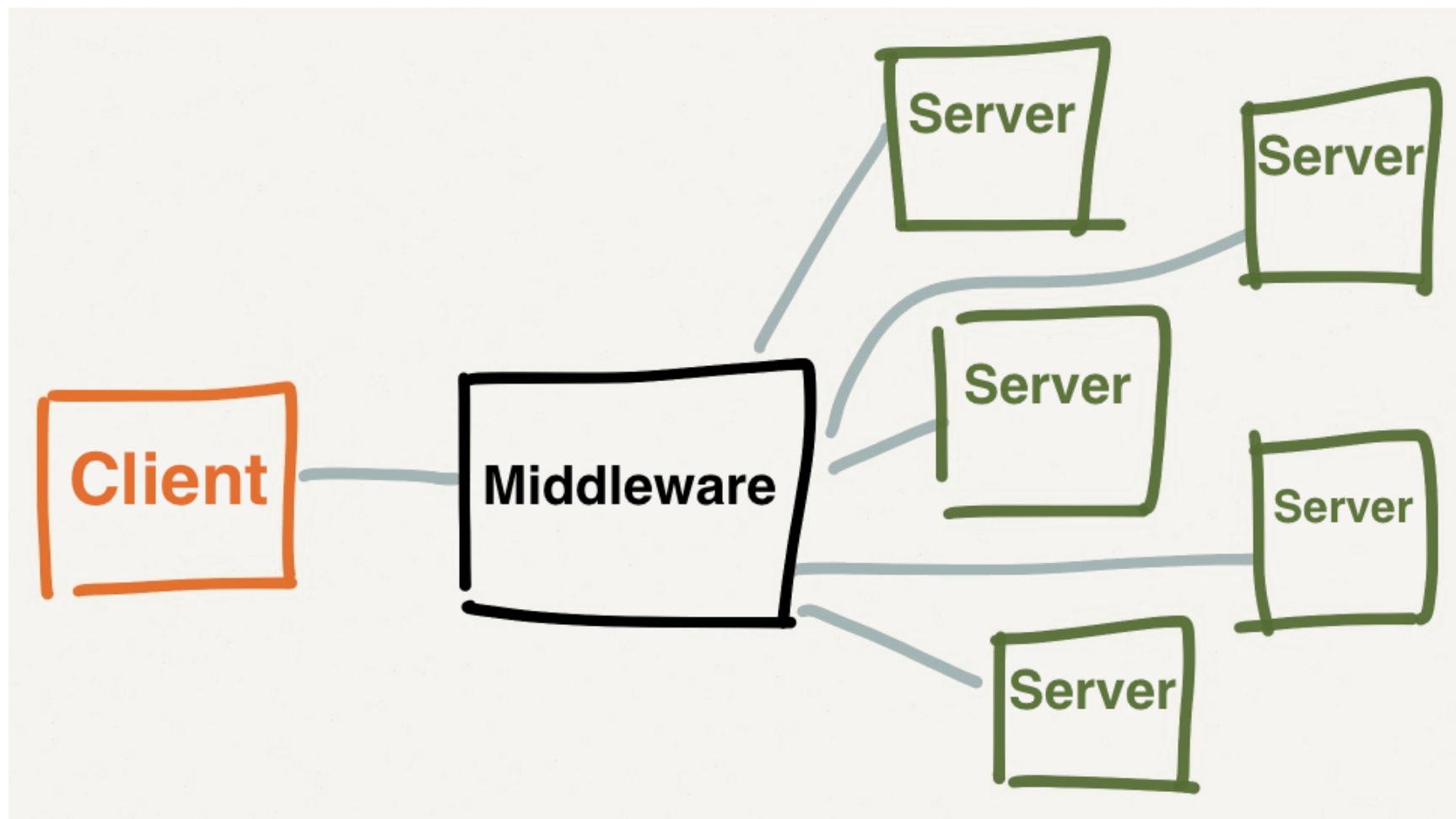
- ▶ Uma aplicação de vendas normalmente precisa de uma interface com sua aplicação de inventário.
- ▶ Uma aplicação de aquisições deve estar conectada a um site de licitações.
- ▶ Uma aplicação deve estar conectada a dispositivos móveis que devem sincronizar com um servidor que mantém um calendário corporativo para que seus funcionários tomem conhecimento dos eventos.

Desafios encontrados na integração

- ▶ Redes são incertas
- ▶ Redes são lentas
- ▶ Aplicações são diferentes
- ▶ Mudanças são inevitáveis



Solução

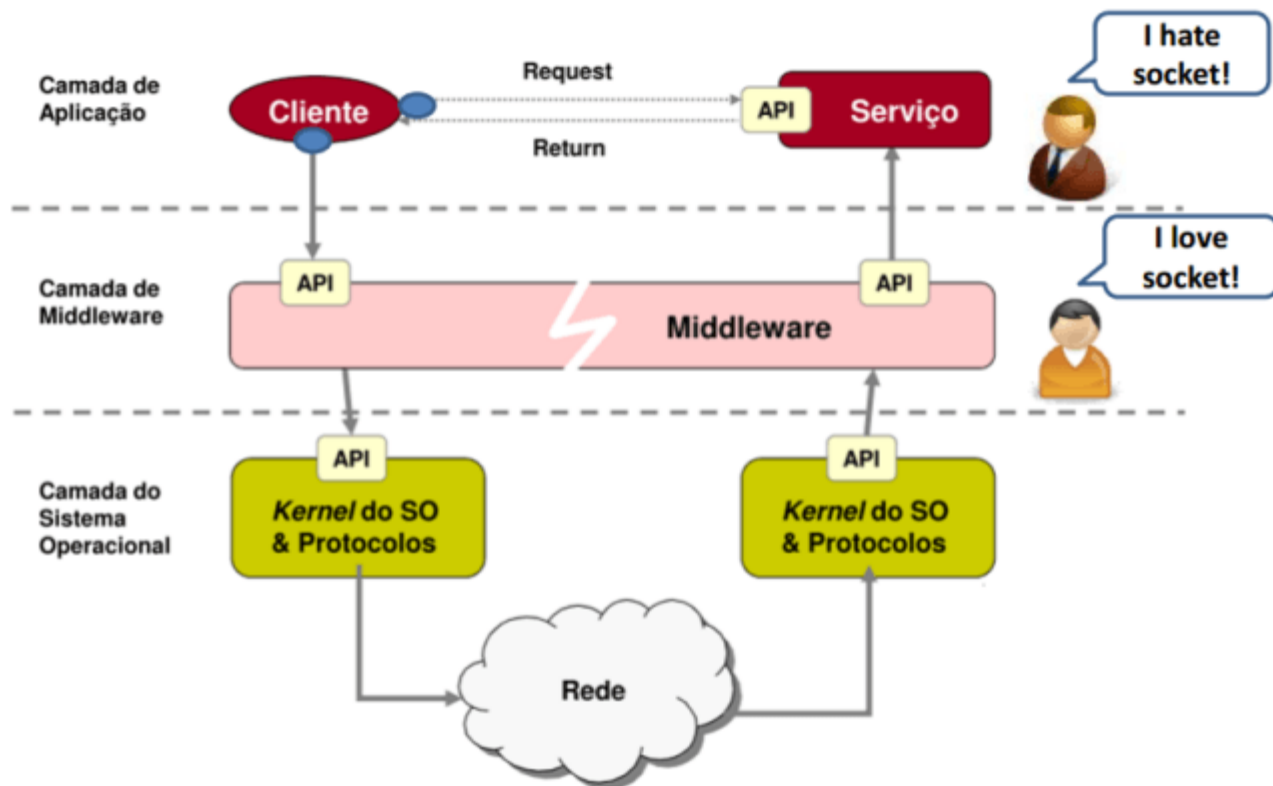


Definições

- ▶ “Um conjunto de serviços que ajuda a resolver problemas de **heterogeneidade** e distribuição” [Bernstein 1996]”.
- ▶ “Software projetado para ajudar a gerenciar a complexidade e a **heterogeneidade** inerente dos sistemas distribuídos” [Bakken 2001]”.
- ▶ “Software localizado entre a aplicação e o SO responsável por resolver a lacuna entre as aplicações e a infraestrutura de software/hardware” [Schmidt 2003]”.



- ▶ Middlewares são mecanismos computacionais que tem a funcionalidade de criar uma camada de mediação entre aplicativos em um ambiente heterogêneo com diferentes tecnologias.



O Middleware reduz o acoplamento entre as aplicações.

Analogia

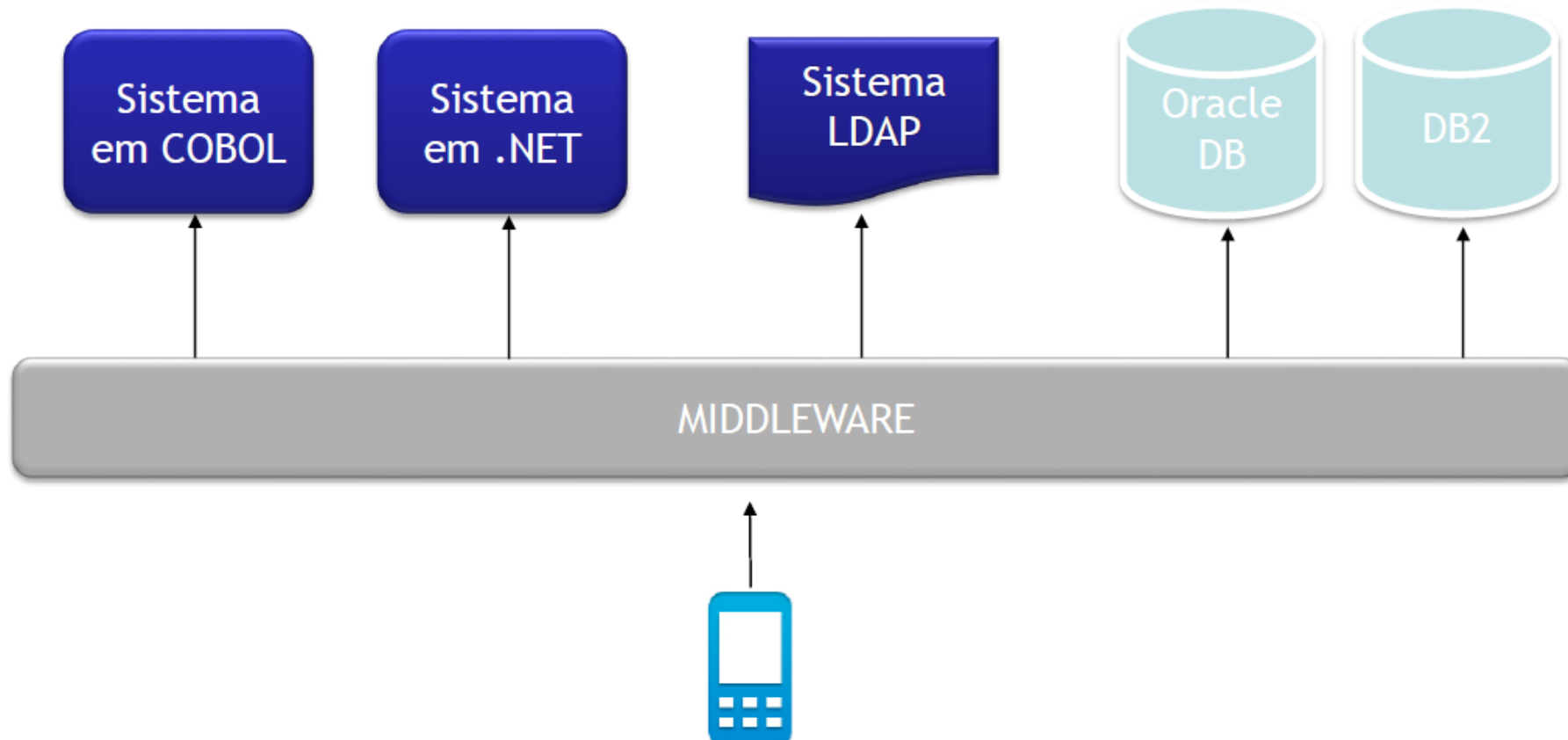


Esse exemplo apresenta a **heterogeneidade** de línguas. Nesses casos, é necessário que haja uma pessoa que conheça as duas línguas para fazer a comunicação fluir através de traduções.

Funções de um Middleware

- ▶ Esconder a Heterogeneidade do Sistema.
- ▶ Prover uma interface uniforme, padronizada e acessível.
- ▶ Prover um conjunto de tecnologias / serviços para funcionalidades gerais.
- ▶ Mascara a distribuição.

Exemplo de um cenário heterogêneo



Serviços de Middleware

As implementações de middlewares provém uma série de serviços que podem ser utilizados, como:

- ▶ -Recursos de comunicação
- ▶ -Persistência
- ▶ -Transações Distribuídas
- ▶ -Segurança

Recursos de comunicação

Suporte a vários protocolos de comunicação e conectores para diferentes tecnologias.

- ▶ RPC(RemoteProcedureCall).
- ▶ Web-Services.
- ▶ Mensageria(Fila de Mensagens).
- ▶ Banco de Dados(Conexão com bancos e chamadas SQL).
- ▶ Conectores para suportar softwares embarcados e de tecnologias fora do universo de TI, como sistemas de automação industrial. (Arduíno, RaspberryPI, etc).

Recursos de persistência

Mecanismos que permitem ao middleware realizar a persistência de informações em diversos mecanismos de armazenamento.

- ▶ Banco de Dados (ORM).



Recursos de Transações Distribuídas

- ▶ Mecanismos que permitem ao middleware gerenciar transações, mesmo que em localidades diferentes. Pense em transação como um conjunto de passos que precisam estar confirmados para que a transação seja considerada completa (**Atomicidade**).
- ▶ Middleware tem a capacidade de gerenciar as partes desta transação e criar um cenário transacional com as partes, permitindo que este cenário gere ações mais ricas e otimizadas para as aplicações/usuários do mesmo.

Recursos de Segurança

Middlewares proveem mecanismos de segurança bastante sofisticados, que permite a gestão de usuários e perfis para acesso aos recursos computacionais disponibilizados pelo middleware. Middlewares de mercado, disponibilizam conectores para várias tecnologias de autenticação/autorização:

- ▶ LDAP
- ▶ Kerberos
- ▶ Java JAAS, Spring Security, dentre outros.

Internet das Coisas





Q1 - [FUNCAB 2014 MDA] Há um termo que se aplica a qualquer software que funciona como uma ligação entre outros dois programas,

como um servidor web ou um programa de banco de dados, também conhecido como um software que atua sendo um intermediário entre

outras aplicações ou dispositivos. Um software com essas características possui a seguinte denominação:

- a) curware.
- b) highware.
- c) appware.
- d) middleware
- e) slowware.

Q1 - [FUNCAB 2014 MDA] Há um termo que se aplica a qualquer software que funciona como uma ligação entre outros dois programas,

como um servidor web ou um programa de banco de dados, também conhecido como um software que atua sendo um intermediário entre

outras aplicações ou dispositivos. Um software com essas características possui a seguinte denominação:

a)curware.

b)highware.

c)appware.

d)middleware

e)slowware.

Q2 - [FCC 2008 TRF5] Um middleware

- a) é um objeto JDBC que encapsula diversos atributos de pesquisa em bancos multiusuários.
- b) está contigualmente localizado entre as camadas de transporte e enlace do modelo OSI.
- c) é uma ferramenta facilitadora para construção de sistemas distribuídos.
- d) é uma ferramenta aplicada na arquitetura de bancos de dados centralizados.
- e) está contigualmente localizado entre as camadas física e de rede do modelo OSI.

Q3 - [CESPE 2012 BASA] O uso de ambiente de TI tem como efeito principal o aumento da interoperabilidade desse ambiente.

Q2 - [FCC 2008 TRF5] Um middleware

- a) é um objeto JDBC que encapsula diversos atributos de pesquisa em bancos multiusuários.
- b) está contigualmente localizado entre as camadas de transporte e enlace do modelo OSI.
- c) é uma ferramenta facilitadora para construção de sistemas distribuídos.
- d) é uma ferramenta aplicada na arquitetura de bancos de dados centralizados.
- e) está contigualmente localizado entre as camadas física e de rede do modelo OSI.

Q3 - [CESPE 2012 BASA] O uso de ambiente de TI tem como efeito principal o aumento da interoperabilidade desse ambiente.
CORRETO.

Sistemas Distribuídos Middlewares orientado a mensagens (MOM)

Prof. Rodrigo Macedo

Motivação

Surge como uma alternativa em relação ao RPC/RMI, para melhorar algumas :

- ▶ Cliente e servidor precisam estar ativos durante a comunicação.
- ▶ Implica em espera para estabelecer o **sincronismo** entre cliente e servidor.
- ▶ **Overhead** para manter conexão / sessão.
- ▶ Falha de uma das partes impede comunicação

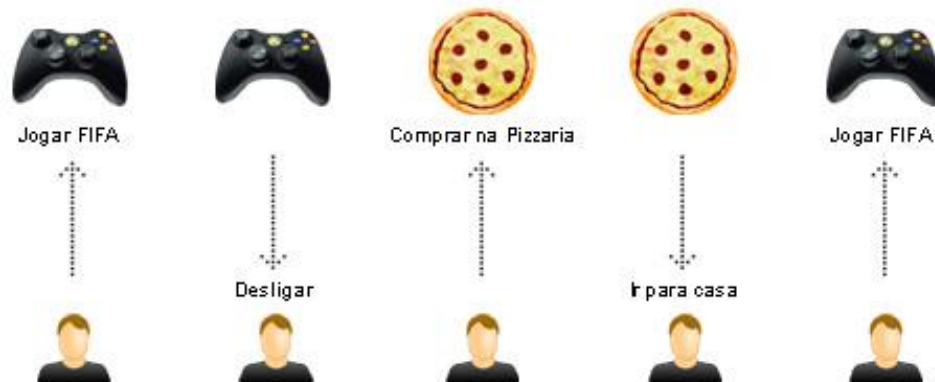


Definição

- ▶ Middlewares orientado a mensagens (MOM's) são middlewares dão suporte extensivo a troca de mensagens **assíncronas**, usando como base mensagens que podem ser enfileiradas e persistidas.
- ▶ Possuem como essência a funcionalidade de prover um meio termo de persistência de mensagens sem a necessidade das partes que irão produzir ou consumir a mensagem estejam **ativas ou conectadas**.
- ▶ As aplicações se comunicam inserindo mensagens em filas.

Síncrono x Assíncrono

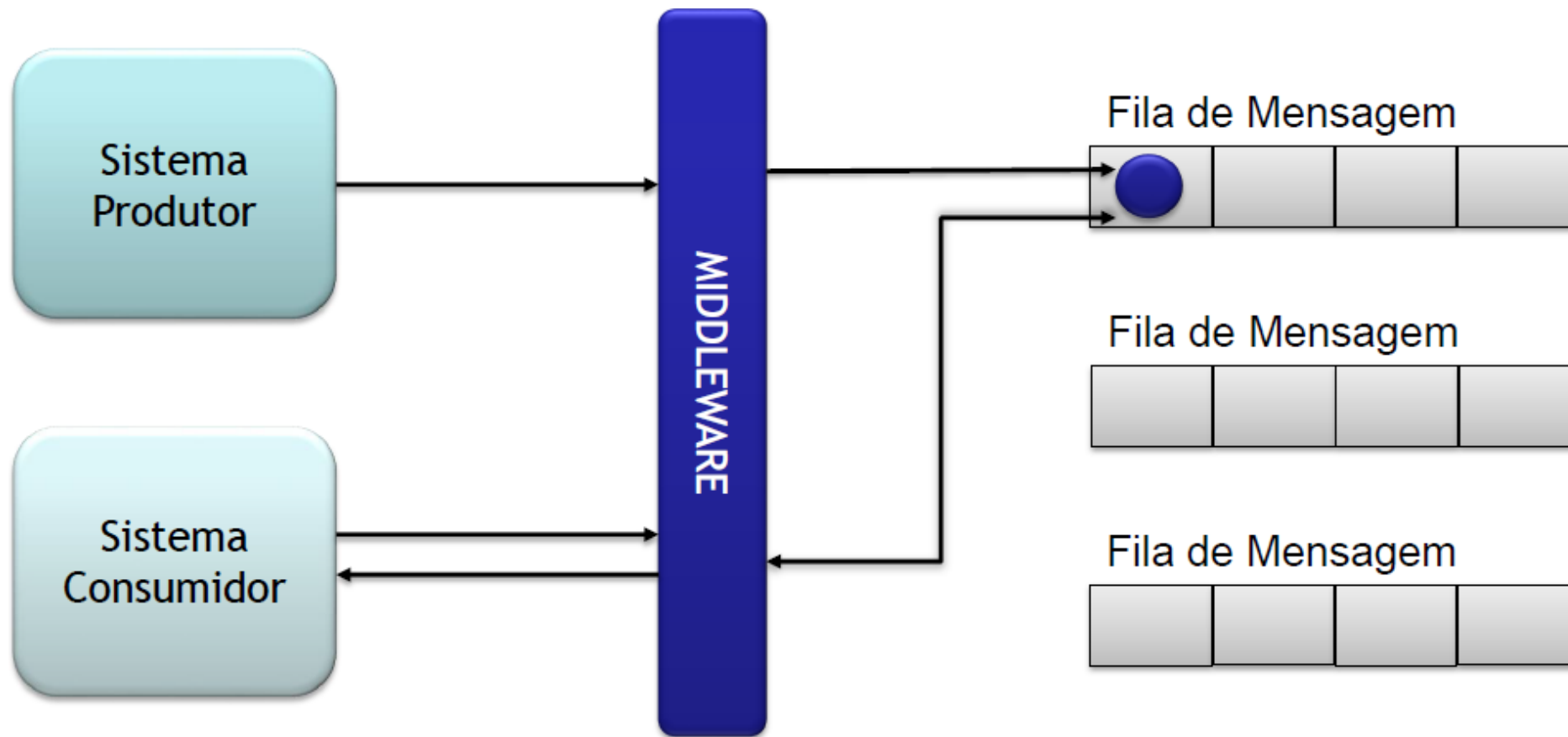
Requisição Síncrona



Requisição Assíncrona



Como funciona



- Mensagem fluxo consumidor - produtor.
- Mensagens em broadcast.

Vantagens



- ▶ Controle padronizado de persistência de informações.
- ▶ Troca de informações assíncronas.
- ▶ Desacoplamento entre aplicações terceiras.
- ▶ Mensagens transmitidas podem conter qualquer tipo de conteúdo.
- ▶ Participantes da comunicação não precisam estar permanentemente conectados à rede (o que gera o overhead).

Desvantagens



- ▶ Dificuldade de manter o sequenciamento das mensagens.
- ▶ Aumento na complexidade da construção e manutenção deste tipo de modelo.
- ▶ Não é recomendado para aplicações em tempo real (por não haver o **sincronismo** entre as mensagens).

Primitivas de Comunicação

- ▶ PUT: adiciona uma mensagem a uma determinada fila
- ▶ GET: obtém uma mensagem de uma certa fila, bloqueando caso a mesma esteja vazia
- ▶ POLL: verifica a fila sem **bloquear**, obtendo uma mensagem caso a fila não esteja vazia
- ▶ NOTIFY: fornece **handler** para ser chamado quando uma mensagem for colocada em uma determinada fila.

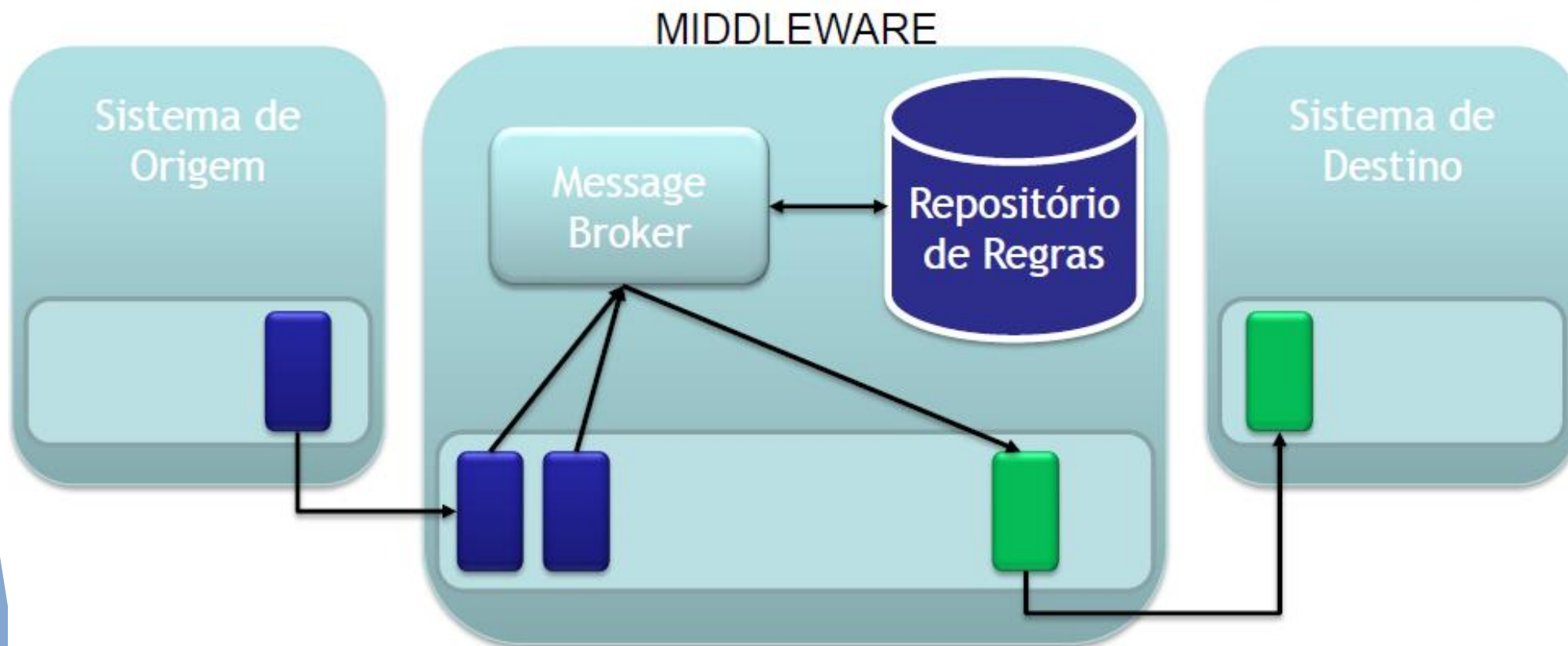
Message brokers

- ▶ Mecanismo que fica dentro do middleware, que tem por responsabilidade **conversão e transformação** da mensagem em um formato que seja legível para o destinatário.
- ▶ Message Brokers possuem um conjunto de regras e algoritmos que auxiliam nesta transformação e no direcionamento da mensagem.
- ▶ É o coração deste tipo de Middleware orientado a mensagem.
- ▶ Atua num ecossistema **heterogêneo**.

Como funciona

<?xml?>

{JSON}



Sistemas Distribuídos API e Web Service

Prof. Rodrigo Macedo

Assuntos

Conceitos que permeiam sobre API's

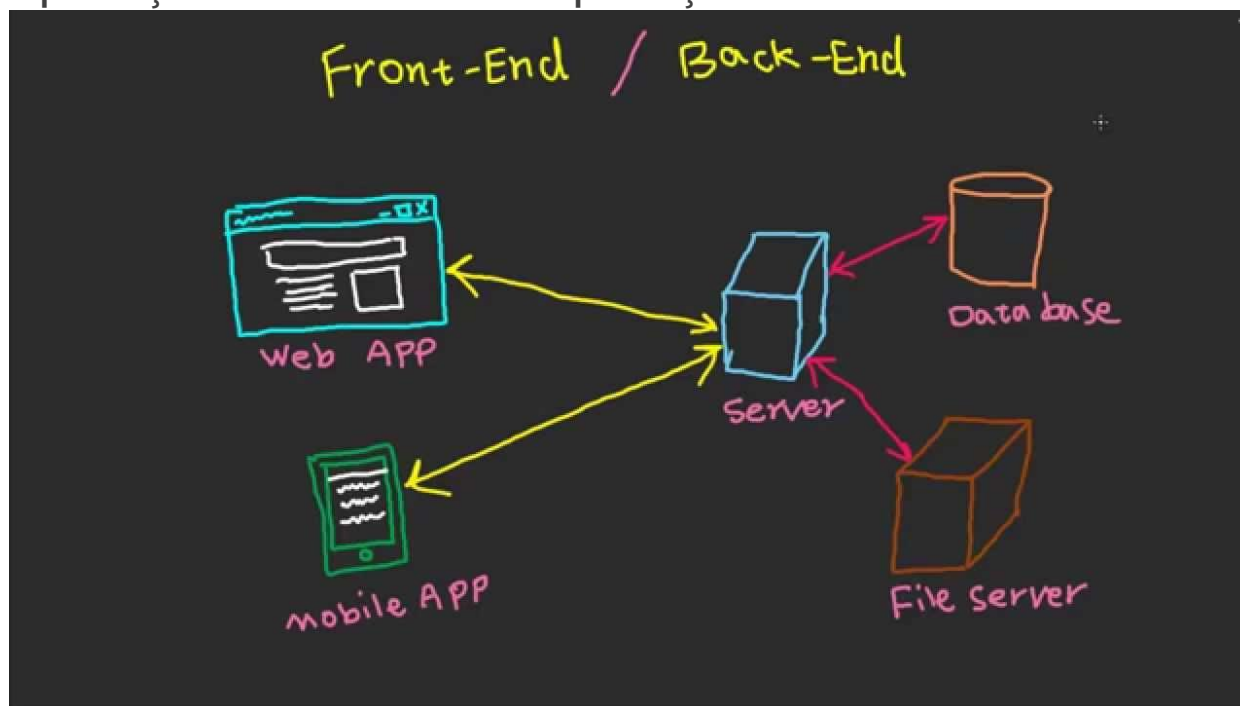
- ▶ Conceitos
- ▶ Motivação
- ▶ Recursos
- ▶ Introdução ao REST.
- ▶ Exemplos



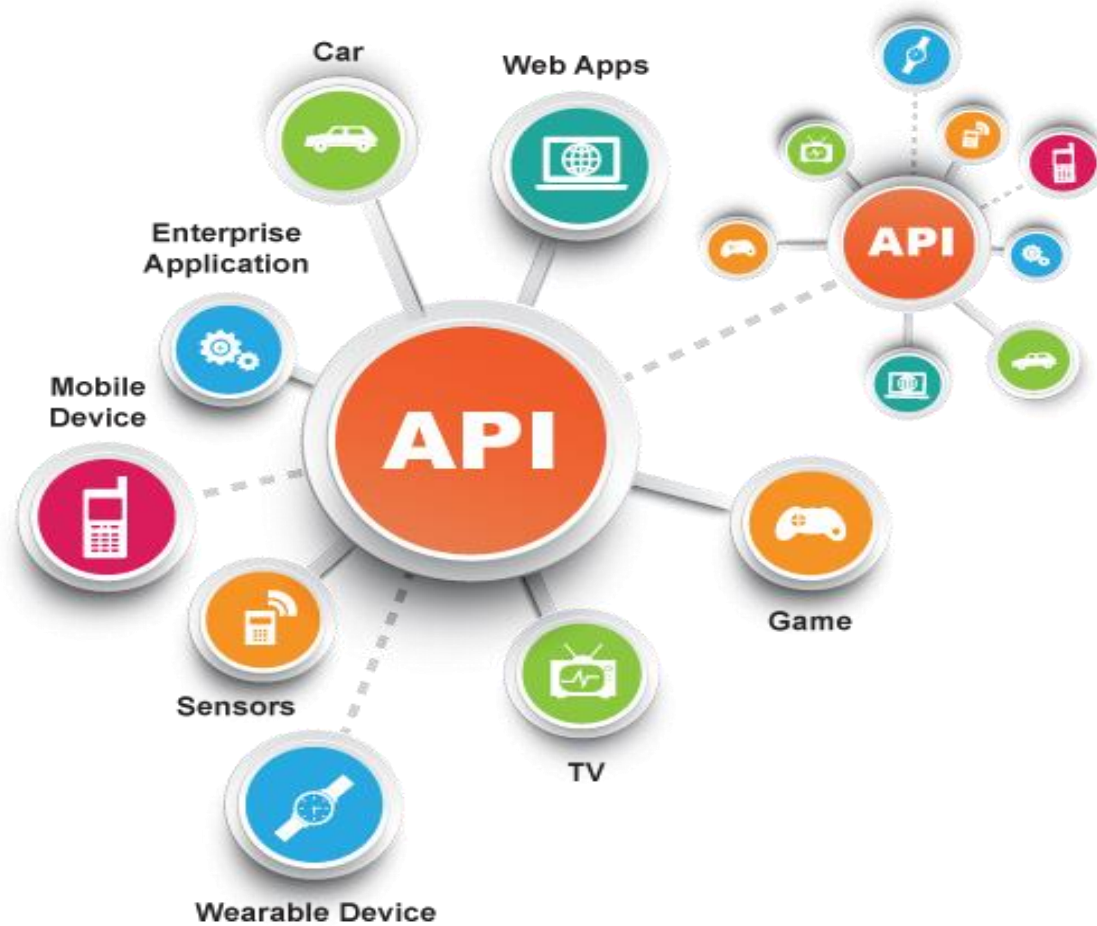
**RESTful
API**

API - Interface de Programação de Aplicativos

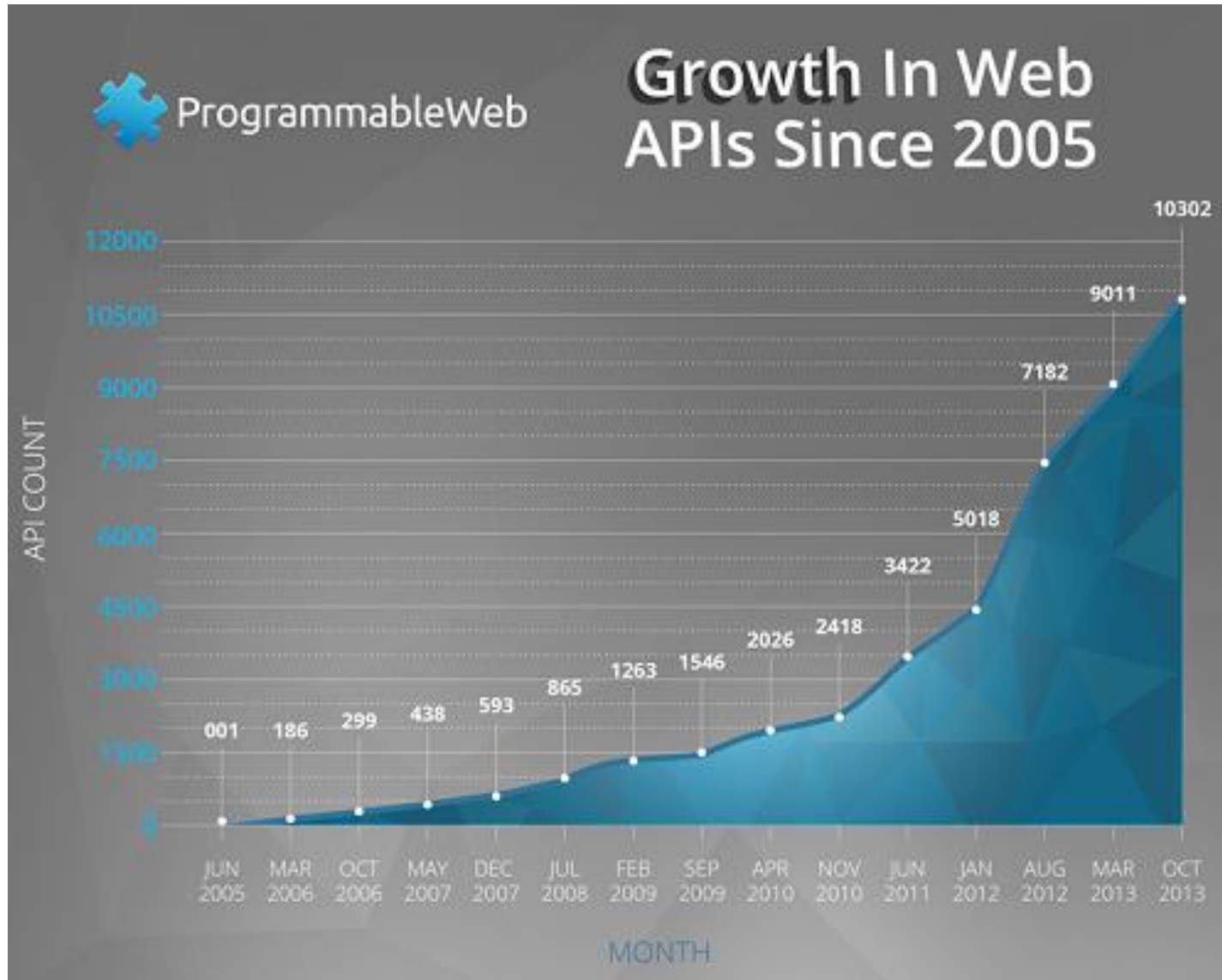
- ▶ Conjunto de rotinas e padrões de programação para acesso a um aplicativo.
- ▶ Com o advento dos client-side frameworks, mais e mais desenvolvedores estão se preocupando em desenvolver o **back-end** que é **compartilhado** entre suas aplicações Web e outras aplicações Nativas



Exemplos



Aplicações Monolíticas x Microserviços



Analogia API X Garçon

Delivery 3444.2425 VACA BURGHER www.vacaburguer.com.br

de Carne

Chaparrada	100g	150g	200g
Hambúrguer e Queijo	11,90	13,90	15,90
Hambúrguer e Bacon	11,90	13,90	15,90
Hambúrguer e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial	11,90	13,90	15,90
Hambúrguer e Molho Especial e Queijo	11,90	13,90	15,90
Hambúrguer e Molho Especial e Bacon	11,90	13,90	15,90
Hambúrguer e Molho Especial e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial e Molho Especial	11,90	13,90	15,90

Agnon

Chaparrada	100g	150g	200g
Hambúrguer e Queijo	11,90	13,90	15,90
Hambúrguer e Bacon	11,90	13,90	15,90
Hambúrguer e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial	11,90	13,90	15,90
Hambúrguer e Molho Especial e Queijo	11,90	13,90	15,90
Hambúrguer e Molho Especial e Bacon	11,90	13,90	15,90
Hambúrguer e Molho Especial e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial e Molho Especial	11,90	13,90	15,90

Alcatra

Chaparrada	100g	150g	200g
Hambúrguer e Queijo	11,90	13,90	15,90
Hambúrguer e Bacon	11,90	13,90	15,90
Hambúrguer e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial	11,90	13,90	15,90
Hambúrguer e Molho Especial e Queijo	11,90	13,90	15,90
Hambúrguer e Molho Especial e Bacon	11,90	13,90	15,90
Hambúrguer e Molho Especial e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial e Molho Especial	11,90	13,90	15,90

Frango

Chaparrada	100g	150g	200g
Hambúrguer e Queijo	11,90	13,90	15,90
Hambúrguer e Bacon	11,90	13,90	15,90
Hambúrguer e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial	11,90	13,90	15,90
Hambúrguer e Molho Especial e Queijo	11,90	13,90	15,90
Hambúrguer e Molho Especial e Bacon	11,90	13,90	15,90
Hambúrguer e Molho Especial e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial e Molho Especial	11,90	13,90	15,90

Lombo

Chaparrada	100g
Hambúrguer e Queijo	11,90

Calabresa

Chaparrada	100g
Hambúrguer e Queijo	11,90

Vegetariano

Chaparrada	100g
Hambúrguer e Queijo	11,90

Australiano

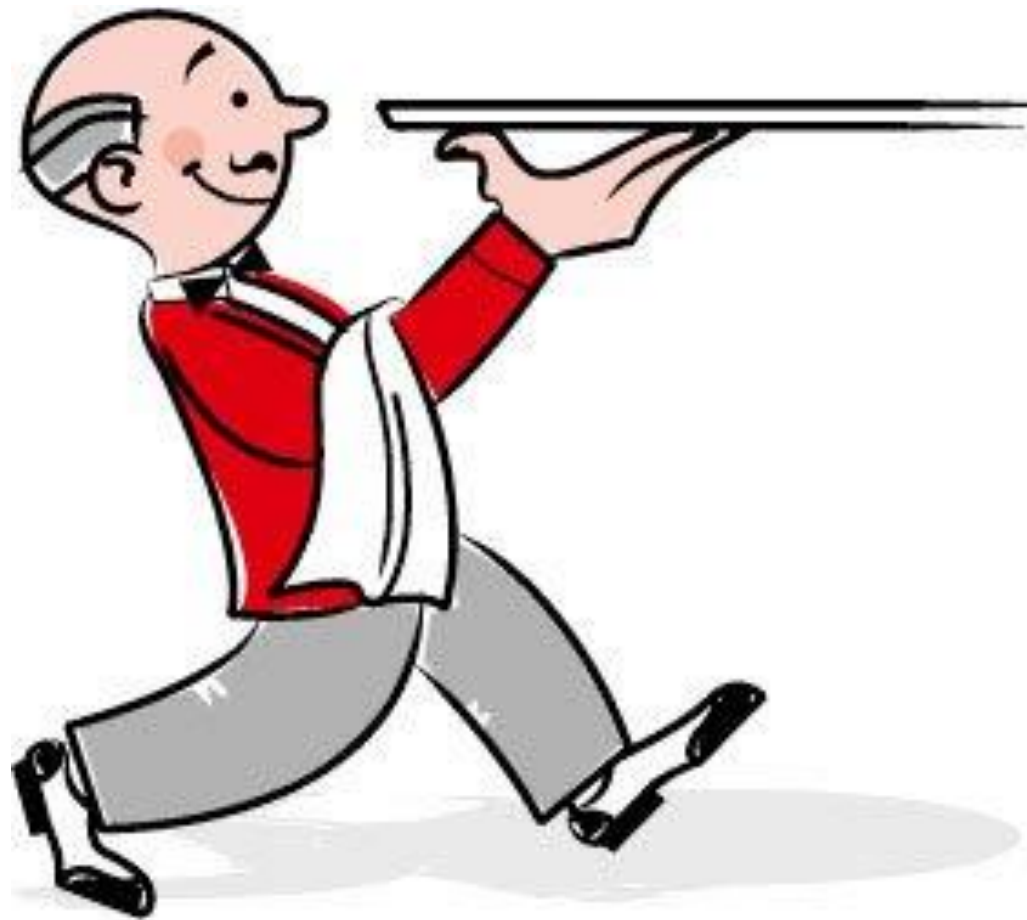
Chaparrada	100g
Hambúrguer e Queijo	11,90

Baguetes

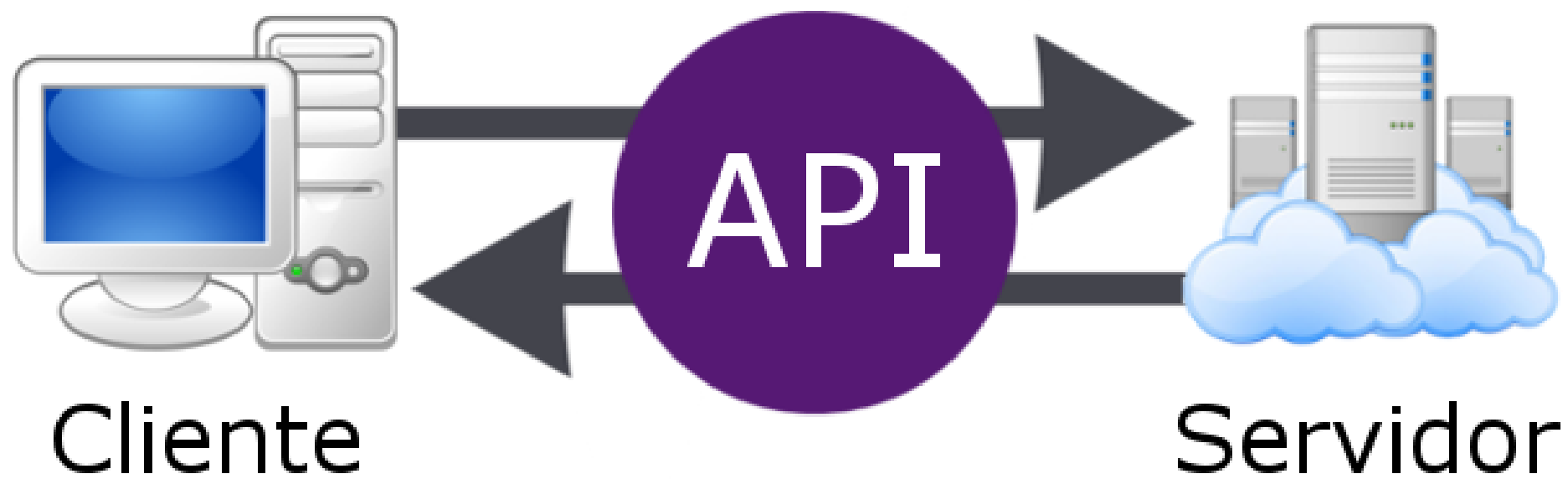
Chaparrada	100g	150g	200g
Hambúrguer e Queijo	11,90	13,90	15,90
Hambúrguer e Bacon	11,90	13,90	15,90
Hambúrguer e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial	11,90	13,90	15,90
Hambúrguer e Molho Especial e Queijo	11,90	13,90	15,90
Hambúrguer e Molho Especial e Bacon	11,90	13,90	15,90
Hambúrguer e Molho Especial e Alho	11,90	13,90	15,90
Hambúrguer e Molho Especial e Molho Especial	11,90	13,90	15,90

Koids

Chaparrada	100g
Hambúrguer e Queijo	11,90
Hambúrguer e Bacon	11,90



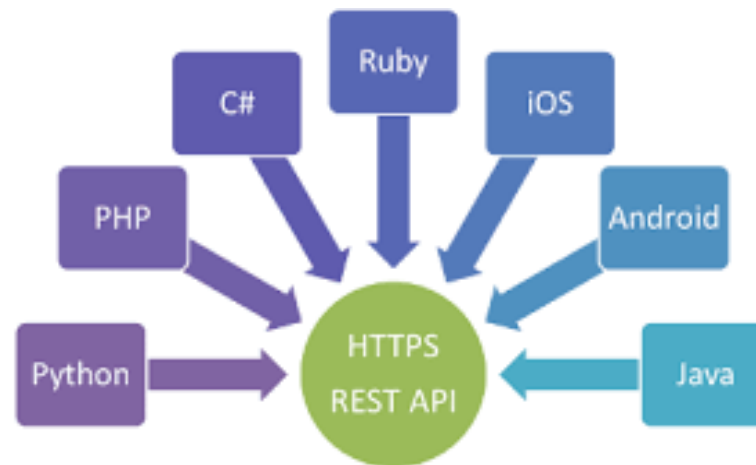
Como funciona



REST



- ▶ “Transferência de dados representacionais”.
- ▶ Introduzido no ano de 2000 numa tese de Ph.D por Roy Fielding, um dos principais autores da especificação do protocolo HTTP.
- ▶ O estilo arquitetural de REST também é aplicado no desenvolvimento de serviços Web. Pode-se caracterizar os web services como "**RESTful**" se eles estiverem em conformidade com as restrições descritas na abordagem REST.



Características

- ▶ **Stateless:** Cada mensagem HTTP contém toda a informação necessária para compreender o pedido. Como resultado, nem o cliente e nem o servidor necessitam gravar nenhum estado das comunicações entre mensagens.
- ▶ **Uso adequado dos métodos HTTP:** Você vai perceber que o Rest está intimamente ligado ao protocolo HTTP.
- ▶ **Código de status:** Outra característica herdada do protocolo HTTP.
- ▶ **Flexibilidade:** Se comparado, ao protocolo SOAP, outro bem difundido em web services, gera menos overhead.
- ▶ **Ausência de um descritor de serviços:** Não ter um 'WSDL' no Rest, como tem no SOAP, conta muito como ponto negativo para o Rest. Em compensação, os desenvolvedores podem estar documentando suas api's através de uma página web.

REST x SOAP

Consider "Martin Lawrence" as your data

SOAP



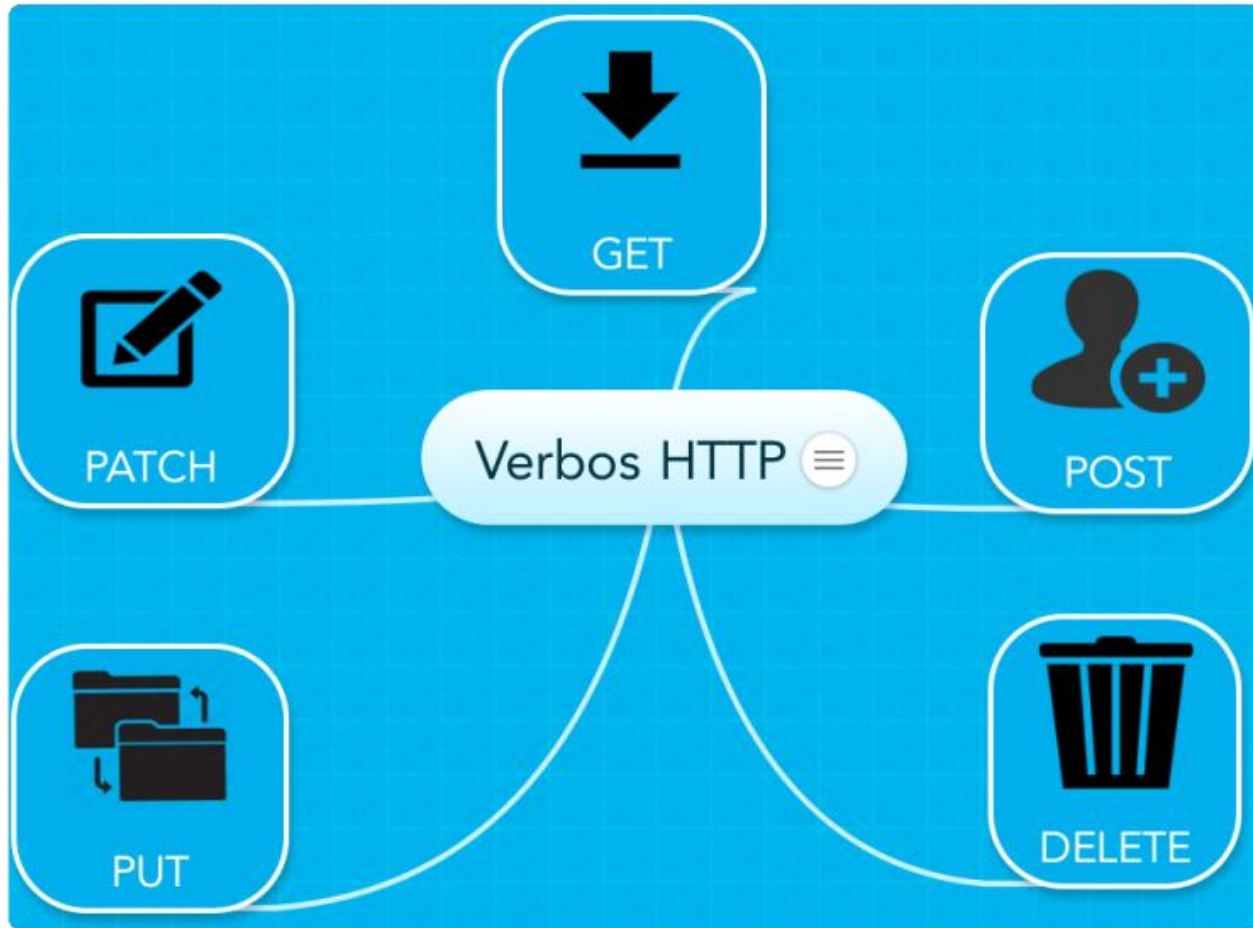
REST



URL's Padronizadas

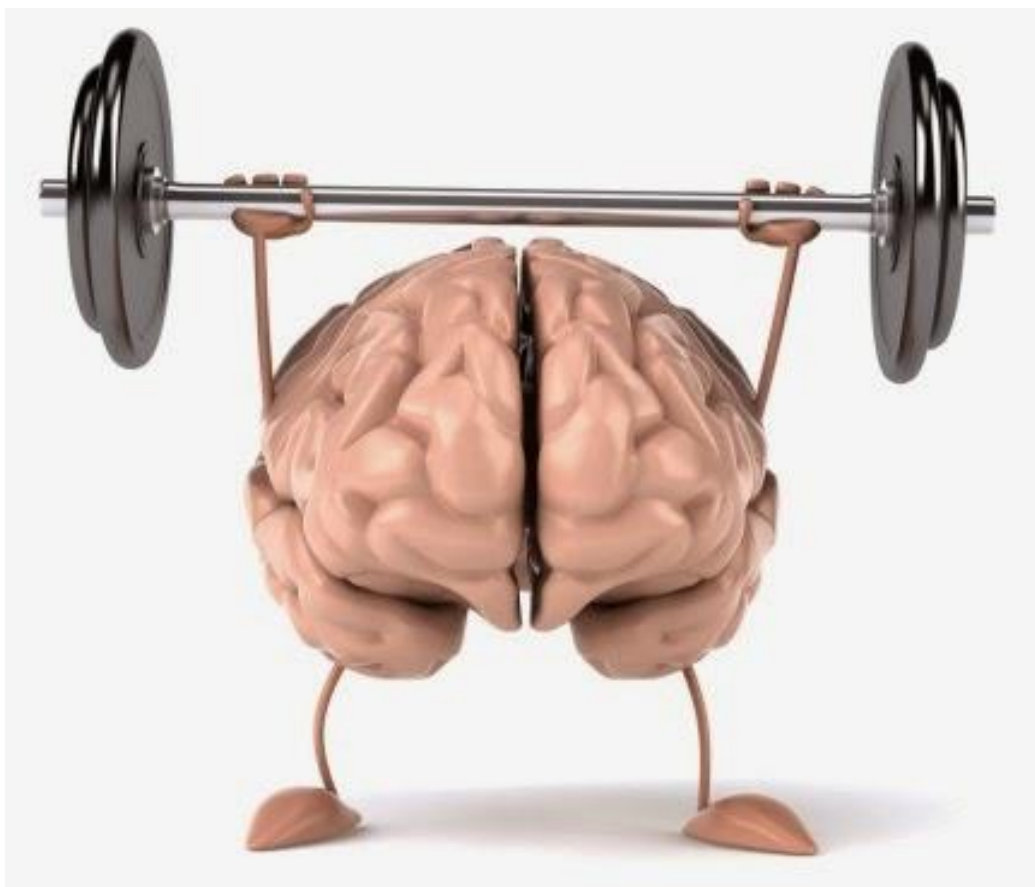
Method	URL	Payload	Result
POST	/api/InventoryItem	CreateInventoryItemCommand (input)	Creates a new inventory item
GET	/api/InventoryItem	InventoryItemListDataCollection (output)	Returns all items
PUT	/api/InventoryItem/{id}	RenameInventoryItemCommand (input)	Renames an item
DELETE	/api/InventoryItem/{id}	DeactivateInventoryItemCommand (input)	De-activates an item
POST	/api/InventoryItem/{id}	RemoveItemsFromInventoryCommand (input)	Removes a number of items from the stock
POST	/api/InventoryItem/{id}	CheckInItemsToInventoryCommand (input)	Adds a number of items to the stock
GET	/api/InventoryItem/{id}	InventoryItemDetail (output)	Returns item details

Métodos HTTP



Status Code

1XX: Informational
2XX: Success
3XX: Redirection
4XX: Client Error
5XX: Server Error



Q1 - [CESPE TRE MT 2015] Sabendo que o código de resposta HTTP (HTTP Response) informa se determinada requisição foi completada com sucesso ou se houve algum problema, sendo a classe da resposta especificada pelo primeiro dígito do código, assinale a opção que apresenta corretamente o nome e a descrição da classe de erro que se inicia com o dígito 5xx.

- A) Redirection: outra ação deve ser executada para completar a requisição
- B) Client Error: a requisição contém erro de sintaxe ou não pode ser completada
- C) Server Error: o servidor não cumpriu uma solicitação aparentemente válida
- D) Informational Request: requisição informacional, o processo continua
- E) Success: a ação foi recebida com sucesso, entendida e aceita

Q2 - [CESPE TRE-GO 2015] Por meio do protocolo chave HTTP, é possível utilizar o método PUT para se criar um novo recurso de um webservice.

Q1 - [CESPE TRE MT 2015] Sabendo que o código de resposta HTTP (HTTP Response) informa se determinada requisição foi completada com sucesso ou se houve algum problema, sendo a classe da resposta especificada pelo primeiro dígito do código, assinale a opção que apresenta corretamente o nome e a descrição da classe de erro que se inicia com o dígito 5xx.

A) Redirection: outra ação deve ser executada para completar a requisição

B) Client Error: a requisição contém erro de sintaxe ou não pode ser completada

C) Server Error: o servidor não cumpriu uma solicitação aparentemente válida

D) Informational Request: requisição informacional, o processo continua

E) Success: a ação foi recebida com sucesso, entendida e aceita

Q2 - [CESPE TRE-GO 2015] Por meio do protocolo chave HTTP, é possível utilizar o método PUT para se criar um novo recurso de um webservice. CERTO.

Q3 - [CESPE TRE-GO 2015] Na busca de um produto em uma loja virtual por meio de um webservice, quando o produto é encontrado, o protocolo HTTP retorna um HTTP/1.1 404, o que facilita o tratamento do pedido no programa cliente.

Q4 - [CESPE MEC 2015] Os valores 200, 301 e 404, no contexto de códigos de resposta do protocolo HTTP, denotam, respectivamente, uma requisição bem-sucedida, um redirecionamento para nova URL e uma página não encontrada.

Q5 - [CESPE INSS 2008] Se o servidor que gerou a página apresentada ficar indisponível, o acionamento do link da região A produzirá um erro do tipo 404 Not Found.

Q3 - [CESPE TRE-GO 2015] Na busca de um produto em uma loja virtual por meio de um webservice, quando o produto é encontrado, o protocolo HTTP retorna um HTTP/1.1 404, o que facilita o tratamento do pedido no programa cliente. **ERRADO.**

Q4 - [CESPE MEC 2015] Os valores 200, 301 e 404, no contexto de códigos de resposta do protocolo HTTP, denotam, respectivamente, uma requisição bem-sucedida, um redirecionamento para nova URL e uma página não encontrada. **CERTO.**

Q5 - [CESPE INSS 2008] Se o servidor que gerou a página apresentada ficar indisponível, o acionamento do link da região A produzirá um erro do tipo 404 Not Found. **ERRADO.**