



Anhanguera

# COMPUTAÇÃO EM NUVEM

tecnologias de suporte à nuvem

Prof. Me Fernando S. Claro  
[fernando.claro@anhanguera.com](mailto:fernando.claro@anhanguera.com)

## ÍNDICE

- 01. Computação em nuvem
- 02. Contexto histórico
- 03. Cluster
- 04. Computação em grade
- 05. Virtualização

## ÍNDICE

**06.** Contêiner de aplicação

**07.** Arquitetura Orientada a Serviços (SOA)

**08.** Exercícios

01.

Computação em nuvem

- Atualmente, a computação em nuvem tem um papel fundamental na indústria de TI, a expectativa é que o mercado de serviços em nuvem continue crescendo nos próximos anos.
- O estudo reportado na revista Forbes (COLUMBUS, 2019) estima um crescimento médio mundial de 12,6% ao ano até 2022, com destaque para serviços IaaS, cuja previsão de crescimento para 2019 é de 27,5% alcançando um volume total de quase 40 bilhões de dólares.

# 02.

## Contexto histórico

- **1990:** Conforme breve histórico descrito em (ERL; PUTTINI; MAHMOOD, 2013), as primeiras iniciativas disponíveis na Internet, para provisionamento remoto de recursos de TI como serviços, surgiram no final dos anos 90. O primeiro caso de sucesso foi o sistema de CRM (*Customer Relationship Management*) (SALESFORCE, 2019) da empresa Salesforce, caracterizado como o primeiro serviço SaaS de sucesso para o mundo corporativo.

- **2002:** Alguns anos depois, em 2002, a Amazon lançaria a plataforma *Amazon Web Services* (AWS) (AMAZON, 2019) que, atualmente, é a líder no mercado de provedores de nuvens públicas para serviços IaaS (DIGNAN, 2019).
- **2006:** O termo computação em nuvem foi usado comercialmente somente em 2006, com o lançamento do serviço IaaS da AWS denominado EC2 (*Elastic Compute Cloud*) (AMAZON, 2019b).
- **2009:** O primeiro serviço PaaS de destaque foi o *Google App Engine* criado em 2009 (GOOGLE, 2019).



- Além dos aspectos históricos, precisamos compreender as tecnologias precedentes, que viabilizaram os modelos de serviços em nuvem. Nesse contexto, é relevante a definição de computação em nuvem como **uma forma de computação distribuída que introduziu modelos de provisionamento remoto de serviços escaláveis** (ERL; PUTTINI; MAHMOOD, 2013).
- Sob essa perspectiva, a computação em nuvem representa uma inovação em relação a outras formas de computação distribuída, entre as quais podemos destacar: computação em cluster e computação em grade.

# 03.

## Cluster

- Um *cluster* é um conjunto de computadores similares interconectados por uma rede de alto desempenho (TANENBAUM; STEEN, 2008).
- O desenvolvimento das tecnologias utilizadas na computação em cluster contribuíram principalmente para o avanço em mecanismos de balanceamento de carga e de recuperação de falhas que também são usados nos provedores de computação em nuvem.

- O que caracteriza a computação em cluster é a homogeneidade, já que todos os computadores do *cluster* devem ter a mesma plataforma (sistema operacional e outros componentes de software) (TANENBAUM; STEEN, 2008).

# 04.

## Computação em grade

- Por outro lado, a computação em grade consiste em reunir recursos computacionais de diferentes organizações, mesmo que as tecnologias de hardware e software sejam diferentes (TANENBAUM; STEEN, 2008).
- Dessa forma, uma grade computacional envolve sistemas computacionais heterogêneos e geograficamente dispersos.



- As tecnologias de computação em grade também influenciaram o desenvolvimento de novos mecanismos que viabilizaram a computação em nuvem, em particular, tecnologias relacionadas com portabilidade das aplicações e gerenciamento de infraestrutura (ERL; PUTTINI; MAHMOOD, 2013).

# 05.

## Virtualização



- Os mecanismos usados em *clusters* e grades computacionais serviram de base para viabilizar os serviços em nuvem.
- Além disso, o crescimento no uso desses serviços também foi fomentado pelo aprimoramento do desempenho das tecnologias de rede, pois o acesso aos serviços é feito através da Internet.

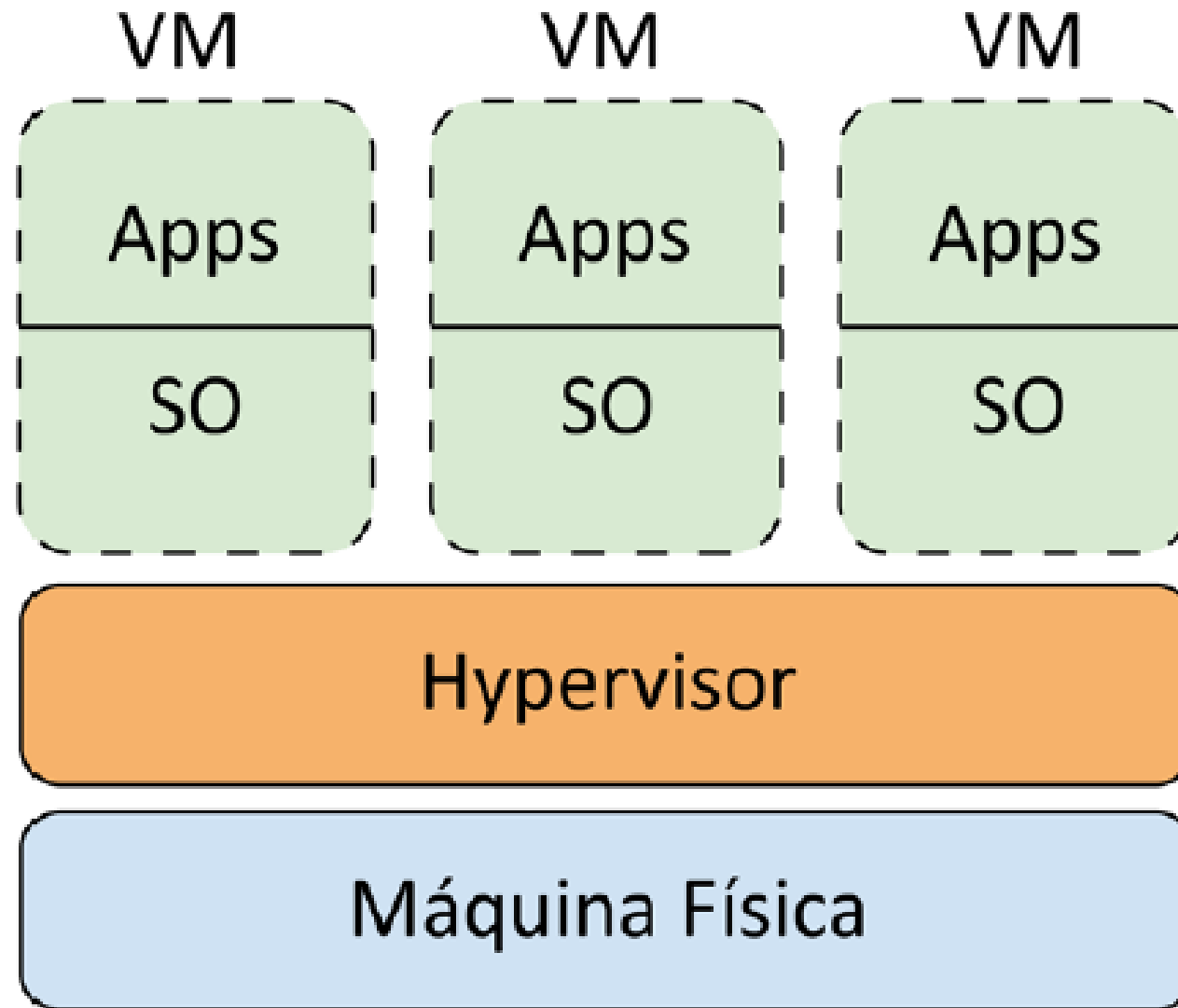
**No entanto, você sabe qual foi a evolução tecnológica chave para o advento da computação em nuvem?**

- Foram os avanços nas tecnologias de virtualização! Como explicado em (ERL, 2013), os softwares de virtualização permitem a criação de múltiplas instâncias lógicas de um recurso computacional de forma que esse recurso possa ser compartilhado entre diversos usuários.
- O conceito de virtualização não é recente, mas, somente com os ganhos em termos de desempenho e confiabilidade das ferramentas de virtualização modernas é que foi possível viabilizar características como a elasticidade rápida e self-service sob demanda, próprias dos serviços de computação em nuvem.

- Foram os avanços nas tecnologias de virtualização! Como explicado em (ERL, 2013), os softwares de virtualização permitem a criação de múltiplas instâncias lógicas de um recurso computacional de forma que esse recurso possa ser compartilhado entre diversos usuários.
- O conceito de virtualização não é recente, mas, somente com os ganhos em termos de desempenho e confiabilidade das ferramentas de virtualização modernas é que foi possível viabilizar características como a elasticidade rápida e self-service sob demanda, próprias dos serviços de computação em nuvem.

- As plataformas de computação em nuvem, como o OpenStack, utilizam as ferramentas de virtualização para gerenciar o compartilhamento de recursos computacionais disponíveis na infraestrutura do ambiente de computação em nuvem.
- Por exemplo, a criação de máquinas virtuais no OpenStack pode ser realizada por meio de diversas ferramentas de virtualização como VMware ESXi, XenServer ou Hyper-V (MICROSOFT, 2019).

- A virtualização pode ser definida como “uma tecnologia que permite criar vários ambientes simulados ou recursos dedicados a partir de um único sistema de hardware físico” (REDHAT, 2019). Como ilustrado na figura a seguir, o ambiente lógico criado sobre o equipamento físico é denominado máquina virtual (VM – *Virtual Machine*).



- A virtualização permite, por exemplo, que um mesmo servidor seja compartilhado por várias aplicações (Apps) de diferentes usuários de forma isolada e segura, pois cada aplicação pode ser executada em uma máquina virtual diferente no mesmo servidor.
- Observe que cada máquina virtual tem seu próprio sistema operacional (SO), então é possível ter máquinas virtuais com sistemas operacionais diferentes no mesmo servidor.

- A virtualização viabiliza três fatores fundamentais para a computação em nuvem: independência de hardware, possibilidade de consolidação de servidores e facilidade de replicação de recursos (ERL, 2013).



- **Independência de hardware:** A ferramenta de virtualização abstrai as peculiaridades dos recursos físicos, de forma que problemas de compatibilidade são minimizados.
- Assim, a migração de uma aplicação em uma máquina virtual não depende das características do hardware do equipamento de destino, desde que o formato da máquina virtual seja suportado pelo hypervisor.

- **Consolidação de servidores:** A consolidação de servidores é um processo para aumentar a taxa de utilização dos servidores em um centro de dados a fim reduzir custos e economizar energia (AHMAD, 2015).
- Uma das formas de consolidação de servidores é migrar as máquinas virtuais para o menor número possível de servidores. Por exemplo, se existe apenas uma máquina virtual em um servidor, ela pode ser migrada para outro servidor que ainda tem recursos disponíveis para que, o primeiro servidor, agora sem nenhuma máquina virtual, possa ser desligado.

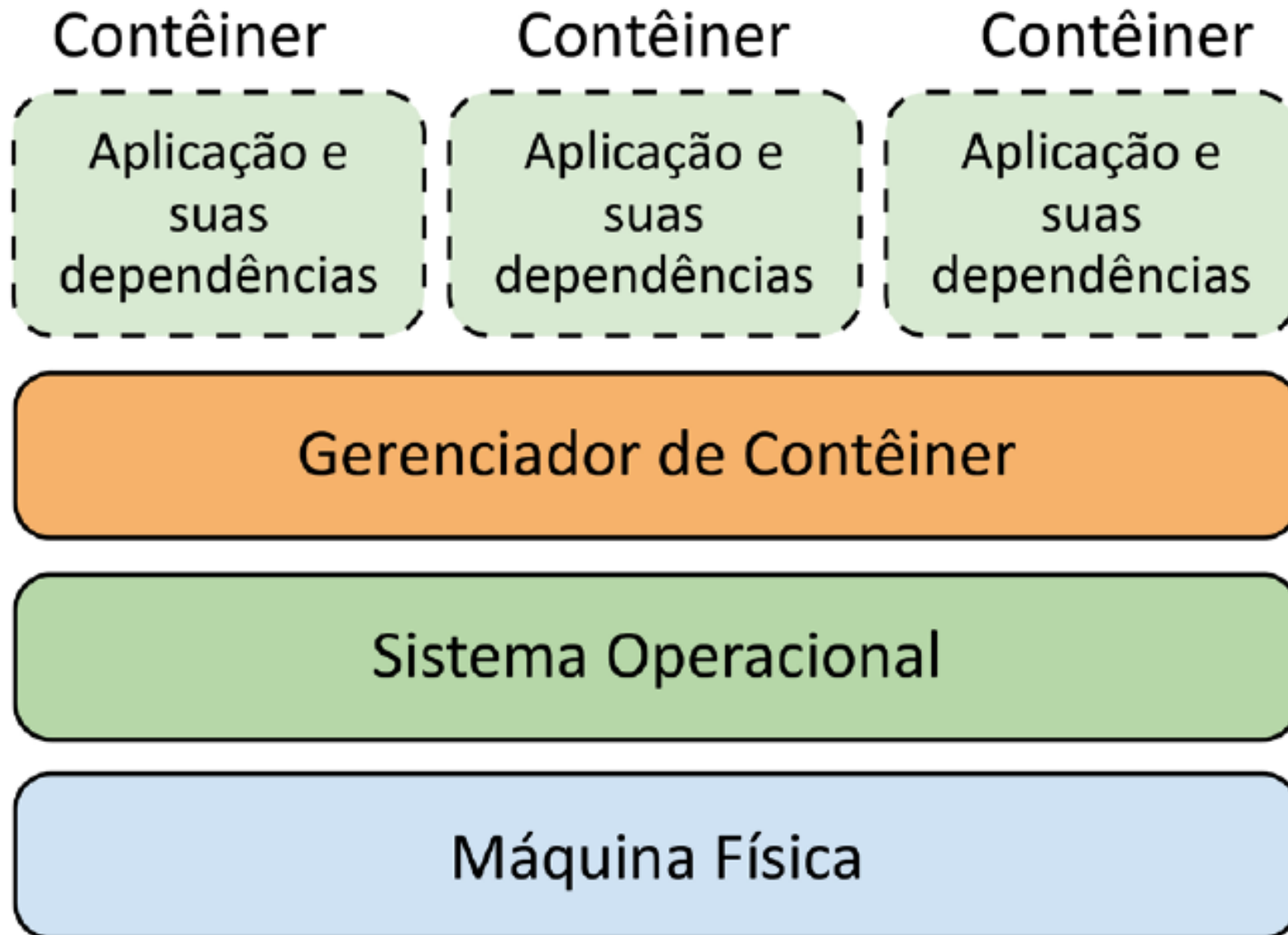
- **Facilidade de replicação:** O terceiro fator importante é a facilidade na replicação das instâncias de máquinas virtuais. Isso decorre do fato de que a máquina virtual é software e pode ser replicada com operações simples de manipulação de arquivos.
- Assim, é mais fácil instanciar e replicar máquinas virtuais do que servidores físicos.

# 06.

## Contêiner de aplicação

- Um modelo alternativo à virtualização baseada em hypervisor é a virtualização baseada em contêiner, que ocorre no nível do sistema operacional (BACHIEGA; SOUZA; BRUSCHI, 2017). Neste caso, um conceito importante é o contêiner de aplicação (*Application Container*) que pode ser entendido como um componente de software autossuficiente, no sentido em que ele encapsula uma aplicação e todas as suas dependências (como bibliotecas, arquivos de configuração, etc.) (SILVA, 2017).

- Diferentemente de uma máquina virtual, o contêiner não inclui um sistema operacional. Na verdade, o sistema operacional do servidor é compartilhado entre os contêineres em execução.
- A figura a seguir ilustra o esquema de contêineres em um servidor físico.
- Com uma ferramenta de gerenciamento de contêineres (*container engine*), pode-se instanciar vários contêineres em uma única máquina física. O hardware e o sistema operacional dessa máquina são compartilhados entre os contêineres (SILVA, 2017).



- A principal característica do contêiner é permitir que o ambiente de execução da aplicação seja sempre o mesmo, pois tudo que a aplicação precisa está encapsulada no contêiner, independentemente da plataforma subjacente (DOCKER, 2019). Isso favorece a portabilidade e facilita a replicação da aplicação em vários servidores.



- Basta, para isso, que um gerenciador de contêiner compatível esteja instalado em cada servidor e uma cópia do contêiner seja copiada e executada em cada um deles.
- Essa agilidade para replicar aplicações ou migrar aplicações de um servidor para outro é muito importante em um ambiente de computação em nuvem, do qual se espera o escalonamento dinâmico de aplicações.

# 07.

## Arquitetura orientada a serviços (SOA)

- Já para a característica de amplo acesso, um fator relevante foi o desenvolvimento de Arquitetura Orientada a Serviços (SOA – *Service Oriented Architectures*) (ERL; PUTTINI; MAHMOOD, 2013).
- Essa arquitetura consiste em decompor as funcionalidades de um sistema em serviços que podem ser reutilizados (CONCEIÇÃO, 2014).



- O objetivo principal desse estilo arquitetural é promover a interoperabilidade entre aplicações.
- Neste caso, os serviços devem ser especificados de forma abstrata, sem dependências em relação a plataformas ou linguagens de programação.

- Além disso, as requisições aos serviços devem ser feitas por meio de tecnologias e padrões abertos. Neste contexto, foram introduzidos os Serviços Web (WS – *Web Services*).
- Esses serviços não são aplicações Web para usuários finais. Eles são componentes de software cujas funcionalidades podem ser invocadas por outras aplicações por meio de requisições HTTP.
- Os dois modelos principais de Serviços Web são: **SOAP Web Services** e **RESTful Web Services** (ERL; PUTTINI; MAHMOOD, 2013).

- O primeiro modelo representou a primeira geração de serviços Web, em que as requisições aos serviços eram especificadas conforme o protocolo denominado SOAP (Simple Object Access Protocol) e encapsuladas em mensagens HTTP.
- As mensagens de requisição SOAP, a descrição dos serviços e a representação dos dados são todas baseadas em esquemas XML (*Extensible Markup Language*) (BECKER, 2001).



- Diferentemente do SOAP, que é um protocolo, o REST (REpresentational State Transfer) é um estilo arquitetural para sistemas distribuídos (RODRIGUES, 2014).
- Os serviços web que seguem os princípios e as restrições REST são então denominados RESTful Web Services.



- O conceito de recurso é fundamental neste modelo de serviço web. Como discutido em (SALVADORI, 2015) sobre os conceitos e princípios do REST, um recurso é qualquer informação, acessível pelo serviço web, que pode ser endereçada através de um identificador padronizado (URI – *Uniform Resource Identifier*) (TANENBAUM; STEEN, 2008, p. 344).



- Os recursos devem ser representados em um formato textual, sendo o mais popular o JSON (*JavaScript Object Notation*) (SALVADORI, 2015).
- Para manipulação dos recursos, são utilizados os métodos padronizados no protocolo HTTP. Os principais métodos são descritos no Quadro 2.1.



- Os recursos devem ser representados em um formato textual, sendo o mais popular o JSON (*JavaScript Object Notation*) (SALVADORI, 2015).
- Para manipulação dos recursos, são utilizados os métodos padronizados no protocolo HTTP. Os principais métodos são descritos no quadro a seguir.



Método	Descrição
GET	Obter a representação textual de um recurso.
POST	Criar um recurso com a representação textual enviada.
PUT	Atualizar os dados de um recurso com a representação textual enviada.
DELETE	Apagar um recurso.



- O estilo arquitetural REST condiciona o uso de tecnologias Web abertas, como o HTTP e o JSON, e o uso de interface padronizada para os serviços, a partir da URI que identifica cada recurso e os métodos HTTP utilizados para manipular esses recursos.
- Esse modelo favorece a interoperabilidade entre as aplicações e a independência em relação a tecnologias e plataformas específicas.



- Esses aspectos são muito importantes para serviços de computação em nuvem, principalmente no que diz respeito à característica de amplo acesso.
- Assim, os RESTful Web Services são uma das principais formas utilizadas pelos provedores de nuvem pública para disponibilizar recursos de TI como serviço para aplicações clientes.

08.

Exercício

- Uma empresa vai lançar um novo aplicativo para gestão de tarefas colaborativas.
- O objetivo é promover aumento de produtividade para pequenas empresas e profissionais liberais.
- O sistema inclui um serviço web e um banco de dados no qual são compartilhados os dados das tarefas.

- O aplicativo se conecta no serviço web por meio de requisições HTTP para manipular os recursos do sistema.
- A empresa está com grande expectativa de sucesso.
- Para atender muitos clientes, já escolheu um grande provedor de nuvem pública para hospedar o sistema.



- Você foi contratado como consultor para definir a melhor estratégia de implantação do serviço web e do banco de dados nesse provedor específico que oferece apenas serviços nos modelos PaaS e IaaS.
- Você precisa determinar qual a melhor opção para implantação do serviço web e do banco de dados em termos do modelo de serviço e tecnologia de virtualização.

# OBRIGADO



Anhanguera