

ARQUITETURA DE SOLUÇÕES EM NUVEM

Augusto Zadra

ARQUITETURAS PARA IMPLEMENTAÇÃO DE PROJETOS EM NUVEM

MINDSET CLOUD NA ENGENHARIA DE SOFTWARE

MUDANÇA DE ARQUITETURA E MICROSERVIÇOS

A MODERNIZAÇÃO CHEGOU NO CÓDIGO

- SERÁ QUE AS APLICAÇÕES ESTÃO PRONTAS PARA ESTE NOVO MODELO DE ARQUITETURA E FUNCIONAMENTO?
- A arquitetura **multi cloud híbrida** oferece um grande poder de decisão para as empresas, equilibrando velocidade, agilidade, custo-benefício e sazonalidade em ambientes e infraestruturas de cloud pública.
- Em paralelo, ao integrar as práticas de **DEVOPS** em uma arquitetura **multi cloud**, as organizações podem obter uma **entrega mais rápida** e consistente de aplicativos em **várias nuvens**, mantendo **altos níveis de confiabilidade e segurança**.

OBJETIVO DE SE ATUALIZAR A ARQUITETURA

- Na maior parte dos negócios inovadores os desenvolvedores criam aplicativos que abrangem funcionalmente o **data center** de uma organização, e adicionam a este projeto uma ou mais **nuvens públicas**.
- Isto permite a utilização de uma variedade de plataformas e modelos de implantação e originou-se uma **variedade de arquiteturas e tecnologias** que os desenvolvedores precisam usar e ferramentas que suportam seu uso.



puppet



CHEF



ANSIBLE



RANCHER®

OBJETIVO DE SE ATUALIZAR A ARQUITETURA

- A utilização do conceito de **containerização** permite que os aplicativos possam ser implantados praticamente em qualquer lugar.
- Sem uma abordagem de **CI/CD** suportada por padrões importantes como o *Kubernetes*, as equipes ficarão atoladas em complexidade.
- Estas tecnologias são OPEN em sua maioria e ajudam a melhorar o ambiente.



puppet



CHEF



ANSIBLE



RANCHER®

MICROSSERVIÇOS

- **MICROSSERVIÇOS** são uma maneira de **projetar** e **desenvolver** aplicativos de software como uma **coleção de serviços pequenos**, independentes e fracamente acoplados.
- A abordagem facilita o desenvolvimento de aplicativos grandes e complexos, além de cada serviço poder ser desenvolvido por uma equipe relativamente pequena de desenvolvedores — ou até mesmo um único desenvolvedor.
- O desenvolvimento paralelo traz na maioria das vezes o maior eficiência no processo e encurta o tempo de desenvolvimento.

MICROSSERVIÇOS

- Os **MICROSSERVIÇOS** são facilitadores de **CI/CD** e mudam as seguintes características:
 - ✓ **Arquitetura** modular descentralizada que caracteriza as implementações baseadas em nuvem que encoraja a **adoção de contêineres**.
 - ✓ Os **contêineres** fornecem um meio de empacotar aplicativos de forma que eles sejam abstraídos de seus ambientes de *run-time*.



MICROSSERVIÇOS

- ✓ No lado dos **desenvolvedores** gasta-se energia na criação da lógica e das dependências do aplicativo.
- ✓ No lado das **operações** concentra-se apenas em como e onde a **lógica é implantada** e como ela é **gerenciada** sem se preocupar com números de versão e problemas de configuração específicos do aplicativo.



MICROSERVIÇOS

- Vejam outros benefícios desta implementação:
 - ✓ **Modularidade:** os microserviços são altamente modulares, o que significa que cada serviço pode ser desenvolvido, implantado e dimensionado independentemente dos outros serviços. Isso facilita a manutenção e atualização do aplicativo ao longo do tempo.



MICROSERVIÇOS

- ✓ **Escalabilidade:** dimensionar de forma independente, com esta regra as arquiteturas de microsserviços são altamente escaláveis e podem lidar com alto tráfego e cargas.
- ✓ **Resiliência:** proporcionam soluções mais resilientes do que as arquiteturas monolíticas, pois falhas em um único microsserviço não afetam todo o aplicativo.



MICROSERVIÇOS

- ✓ **Agilidade:** permitem que as organizações liberem novos recursos e atualizações com rapidez e frequência, pois cada microsserviço pode ser desenvolvido e implantado de forma independente.
- ✓ **Diversidade de tecnologia:** permitem a diversidade de tecnologia, pois cada microsserviço pode ser desenvolvido usando uma pilha de tecnologia diferente, se necessário.



EVOLUÇÃO COMPLETA

1990s and earlier
Coupling

Pre-SOA (monolithic)
Tight coupling



2000s

Traditional SOA
Looser coupling



2010s

Microservices
Decoupled



Arquiteturas de microserviços também apresentam alguns **desafios**, como **maior complexidade na comunicação entre os serviços**, **difículdade de teste e monitoramento** e a necessidade de uma **cultura DEVOPS** madura para gerenciar as muitas partes móveis da arquitetura.

DISCURSOS ADERENTES! CLOUD E MICROSERVIÇOS. VAMOS FALAR SOBRE NATIVIDADE!

REFERÊNCIAS BIBLIOGRÁFICAS

DOCKER, 2022. Use containers to Build, Share and Run your applications. Disponível em: <<https://www.docker.com/resources/what-container/>>. Acesso em: 13 Ago. 2022.

KRESS, Jürgen et al. SOA and Cloud Computing 2014. Disponível em: <<https://www.oracle.com/technical-resources/articles/middleware/soa-ind-soa-cloud.html>>. Acesso em: 15 Ago. 2022.

TAURION, Cesar; Grid Computing : um novo paradigma computacional. Rio de Janeiro: BRASPORT, 2004.

VERAS, Manoel. Cloud Computing: nova arquitetura da TI. 1ª ed. Rio de Janeiro: Brasport, 2012, 214 p.

VERAS, Manoel. Datacenter: Componente Central da Infraestrutura de TI, Brasport, 2009.



PUC Minas
Virtual