

ARQUITETURA DE SOLUÇÕES EM NUVEM

Augusto Zadra

ARQUITETURAS PARA IMPLEMENTAÇÃO DE PROJETOS EM NUVEM

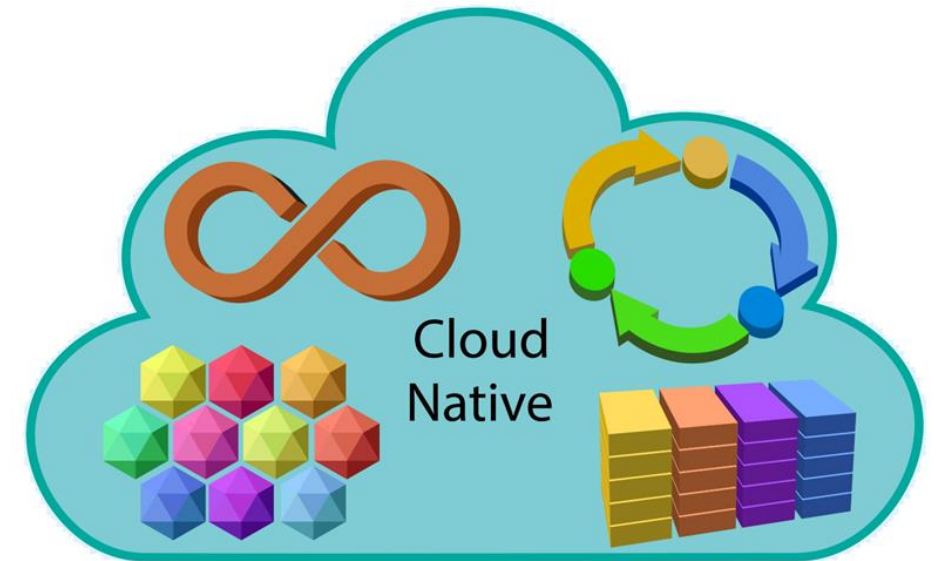
COMPONENTES FUNCIONAIS PARA IMPLANTAÇÃO DE APLICAÇÕES *CLOUD NATIVE*

CLOUD NATIVE

- O QUE INTERESSA AO ARQUITETO DE SOLUÇÕES?
- À medida que a nuvem amadurece e se torna mais sofisticada, ela também evolui para dar suporte à forma como os **aplicativos** em nuvem são definidos fica claro o caminho **topológico** a seguir **independente do negócio** e as facilidades que a inserção de diversas ferramentas irão nos trazer.
- Aplicativos nativos de nuvem são aplicativos projetados e desenvolvidos especificamente para implantação em plataformas de nuvem e buscam aproveitar os benefícios das plataformas de nuvem, como **escalabilidade, disponibilidade e resiliência**.

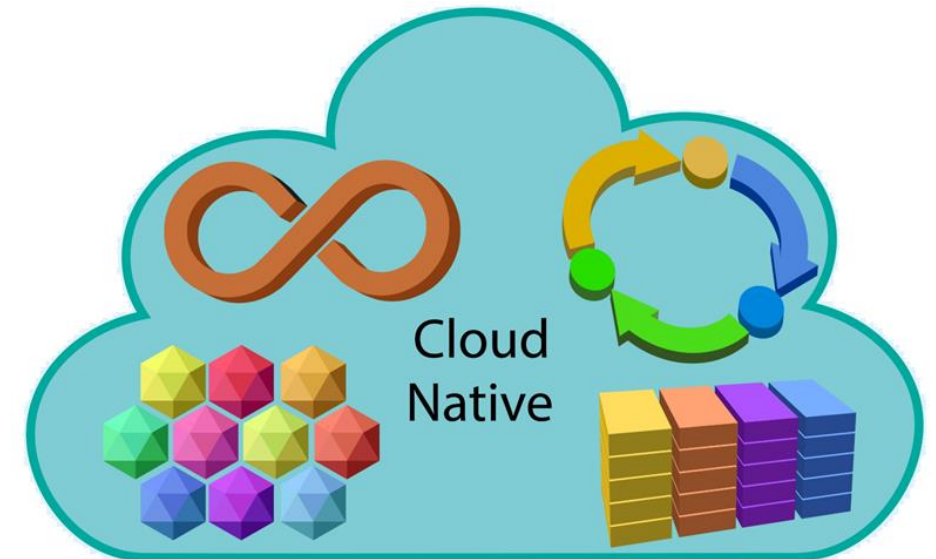
CARACTERÍSTICAS DESTA ARQUITETURA

- Aplicativos *Cloud Native* são ofertas de software projetadas com **microsserviços**, **contêineres** e **orquestração dinâmica**, bem como **entrega contínua de software**.
- Cada parte do aplicativo *Cloud Native* é armazenada em seu próprio **container** e **orquestrada dinamicamente** com outros **containers** otimizando a forma como os recursos são utilizados.



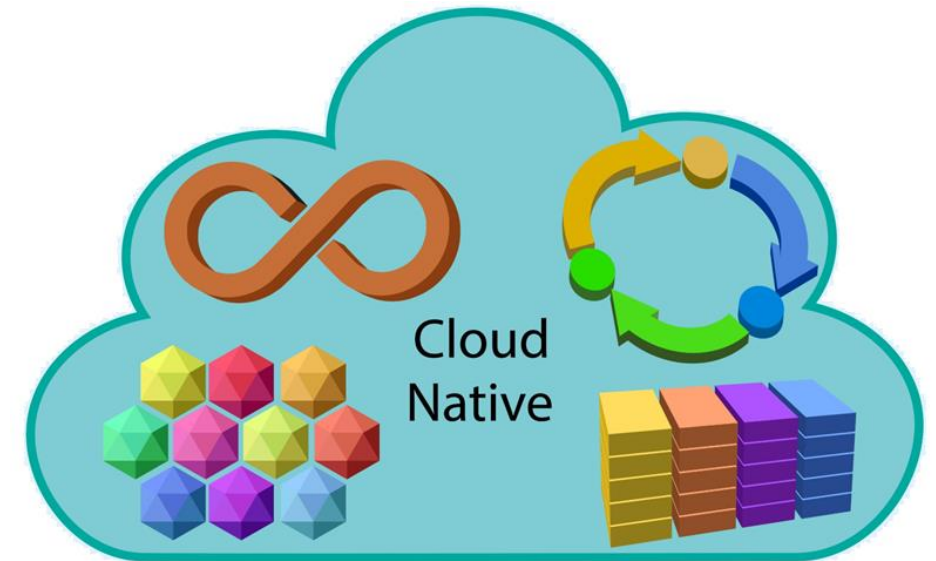
EM RESUMO...

- Compilando todas as informações aplicações *cloud native* rodam em ambientes containerizados, implementando microsserviços, com ferramentas e processos de **DEVOPS**, e **AGNÓSTICO DE PLATAFORMA** - o que significa que podem ser implantados em qualquer plataforma de nuvem sem a necessidade de fazer alterações significativas no código do aplicativo.



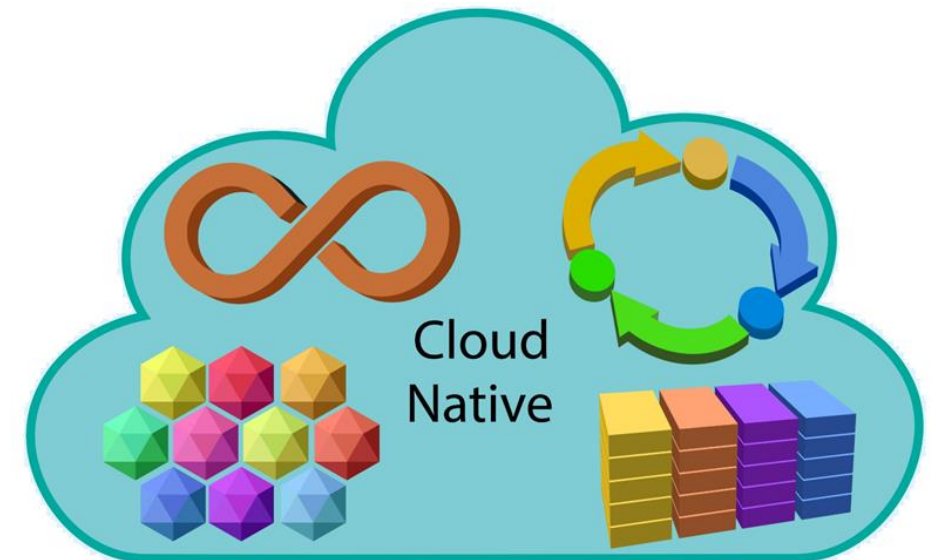
CARACTERÍSTICAS DESTA ARQUITETURA

- São aplicativos que possuem uma abordagem de **microsserviços** decompondo um **monolito** em **serviços modulares** e independentes que interagem por meio de **contratos** de serviço bem definidos.
- Esta arquitetura é apropriada para sistemas implantados na infraestrutura em nuvem e plataforma de containers, pois pode tirar proveito da **elasticidade** e **recursos de provisionamento sob demanda**.
- **MAS O QUE SERIAM ESTES CONTRATOS?**



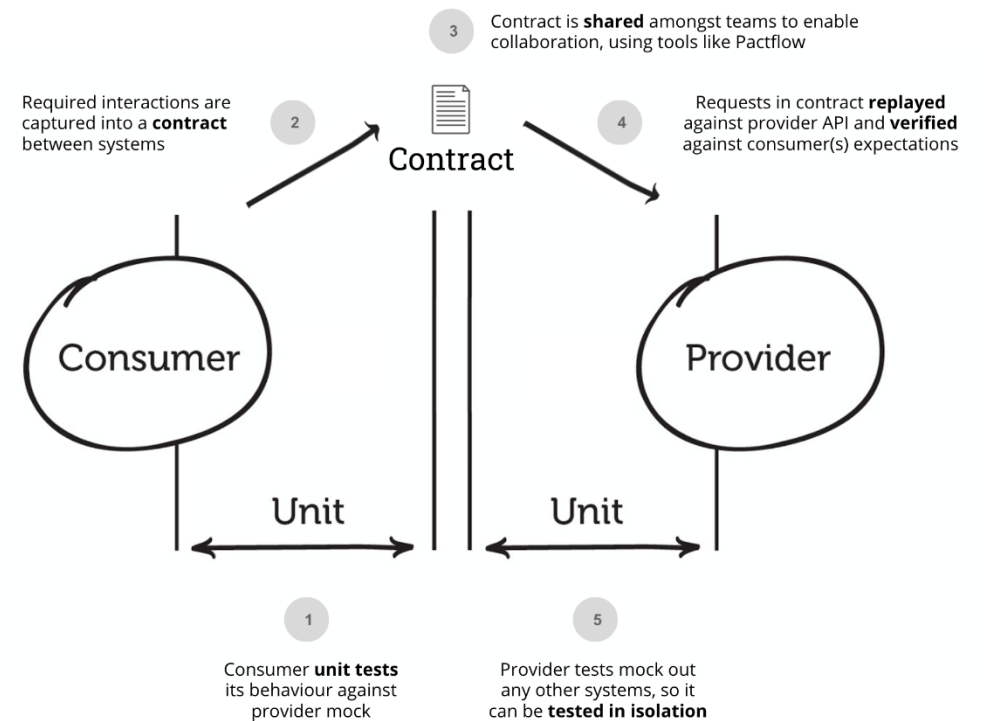
CARACTERÍSTICAS DESTA ARQUITETURA

- São aplicativos que possuem uma abordagem de **microsserviços** decompondo um **monolito** em **serviços modulares** e independentes que interagem por meio de **contratos** de serviço bem definidos.
- Esta arquitetura é apropriada para sistemas implantados na infraestrutura em nuvem e plataforma de containers, pois pode tirar proveito da **elasticidade** e **recursos de provisionamento sob demanda**.
- **MAS O QUE SERIAM ESTES CONTRATOS?**



AS REGRAS DE INTEGRAÇÃO

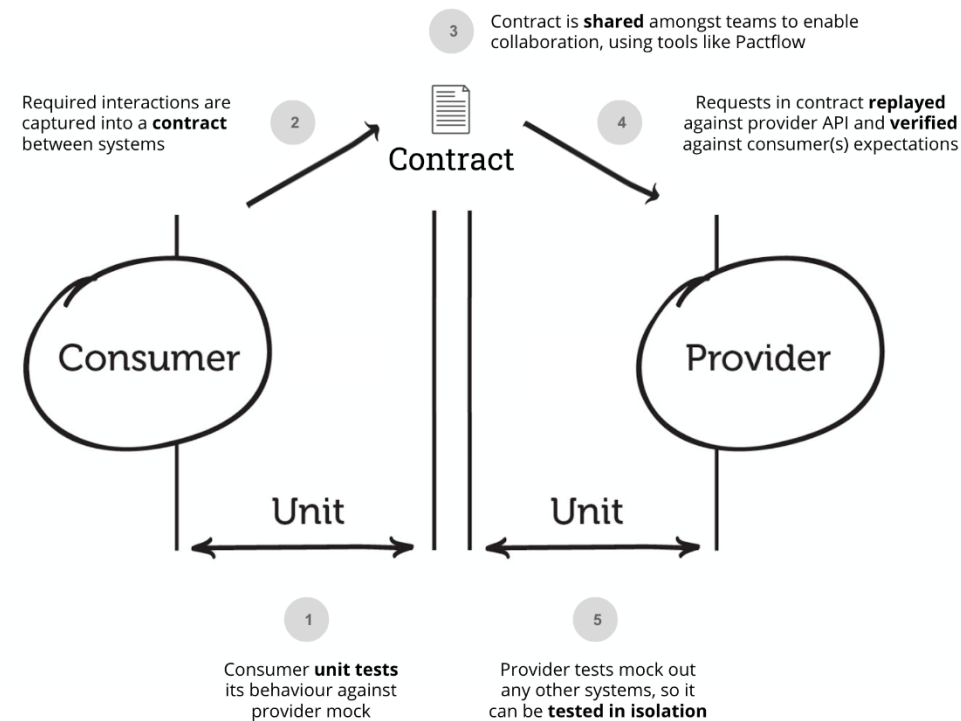
- Microserviços trazem uma ideia de fornecer ambientes extremamente objetivos e de tarefas específicas.
- A API ofertada geralmente terá apenas um função objetiva, o que representa um **contrato público**.
- Vamos por exemplo pensar em uma **API** que fornece o nome de um cliente, número de telefone e endereço quando receberem o ID do cliente. **Este é o contrato e não muda!**



Fonte: <https://docs.pact.io/>

AS REGRAS DE INTEGRAÇÃO

- Simplificando o entendimento, um contrato é uma coleção de acordos entre um cliente (*Consumer*) e uma API (*Provider*) que descreve as interações que podem ocorrer entre eles.
- Existem ferramentas que fazem este teste em diversas linguagens.
- Mas para que eu preciso disto?
- **Por questões de lógica de negócio!**



Fonte: <https://docs.pact.io/>

CONTRATOS EM NUVEM

- **E o que isto tem a ver com a contratação da nuvem?**
- Cada microsserviço precisa fornecer um contrato bem definido, com controle de versão, aos clientes, que são outros microsserviços.
- O serviço não pode quebrar esses contratos com controle de versão até que se tenha certeza de que nenhum outro microsserviço dependa de um determinado contrato com controle de versão.
- Os serviços que você contrata nos provedores de cloud para consumo e otimização de aplicações e serviços são consumidos.

A NOVA LINHA DE RACIOCÍNIO

- Tradicionalmente, muitas organizações consideravam a nuvem por **causa dos custos mais baixos** – uma razão válida, mas limitada porém, isto foi antes de terem a visibilidade que poderiam criar aplicativos ***cloud native***.
- A partir deste ponto a economia de custos **deixa de ser o foco exclusivo** das empresas que passa a ser a **capacidade de criar aplicativos rapidamente** e trazer vantagem competitiva para os negócios.



A NOVA LINHA DE RACIOCÍNIO

- Os aplicativos *cloud native* são criados para serem executados em hardware **modular** e **automatizado**, permitindo que eles se tornem **resilientes** e **previsíveis**.
- Desempenho e escalabilidade tornam-se benefícios importantes, resultantes da capacidade da implantação de cargas de trabalho de forma flexível onde quer que elas estejam e **os aplicativos tradicionais simplesmente não oferecem esses benefícios.**



OS REQUISITOS PARA RODAR NA INTERNET ESTÃO POSTOS! E A COMUNICAÇÃO?

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