



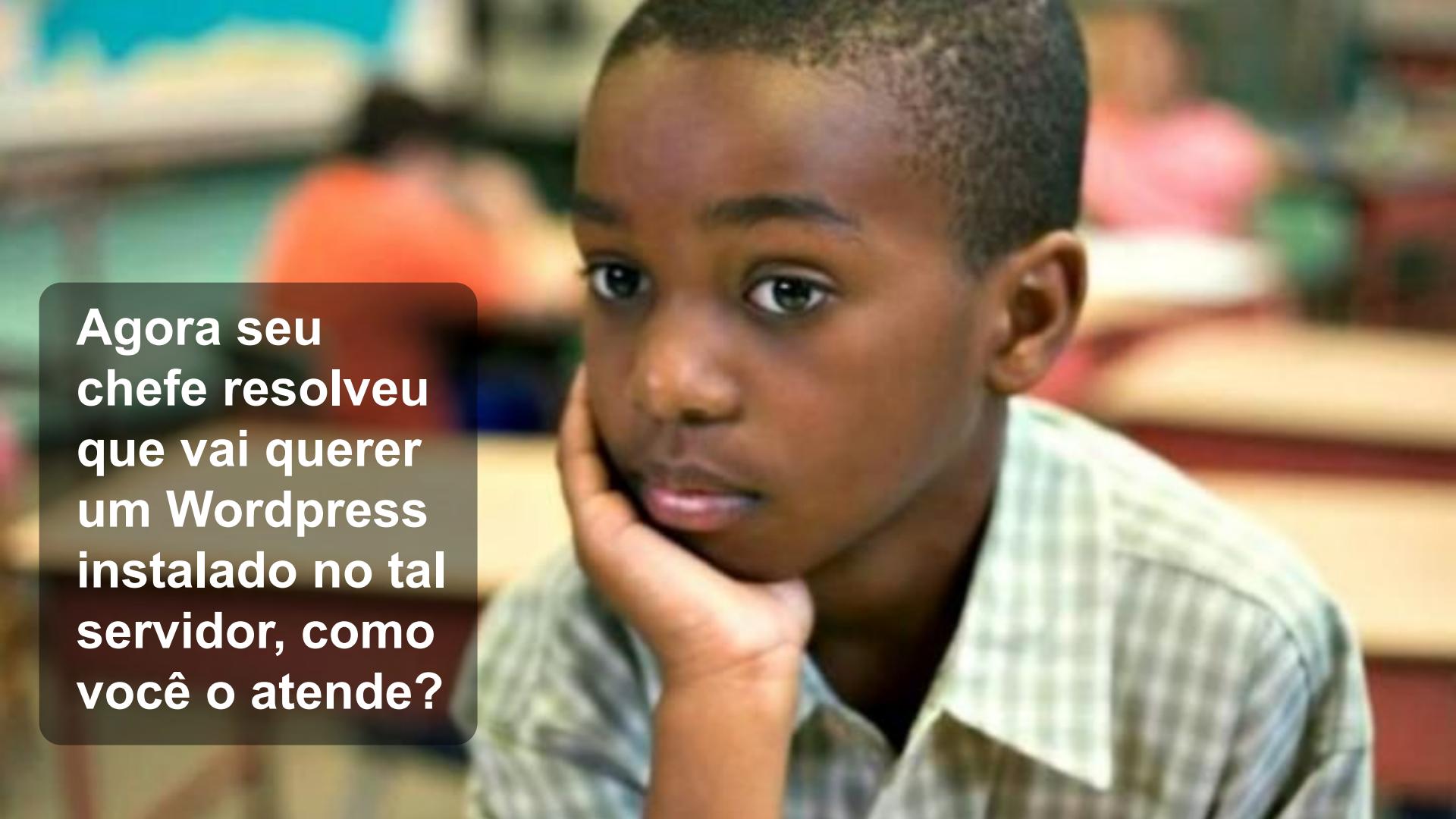
Orquestradores de Nuvem

Leonardo Rebouças de Carvalho
leouesb@gmail.com





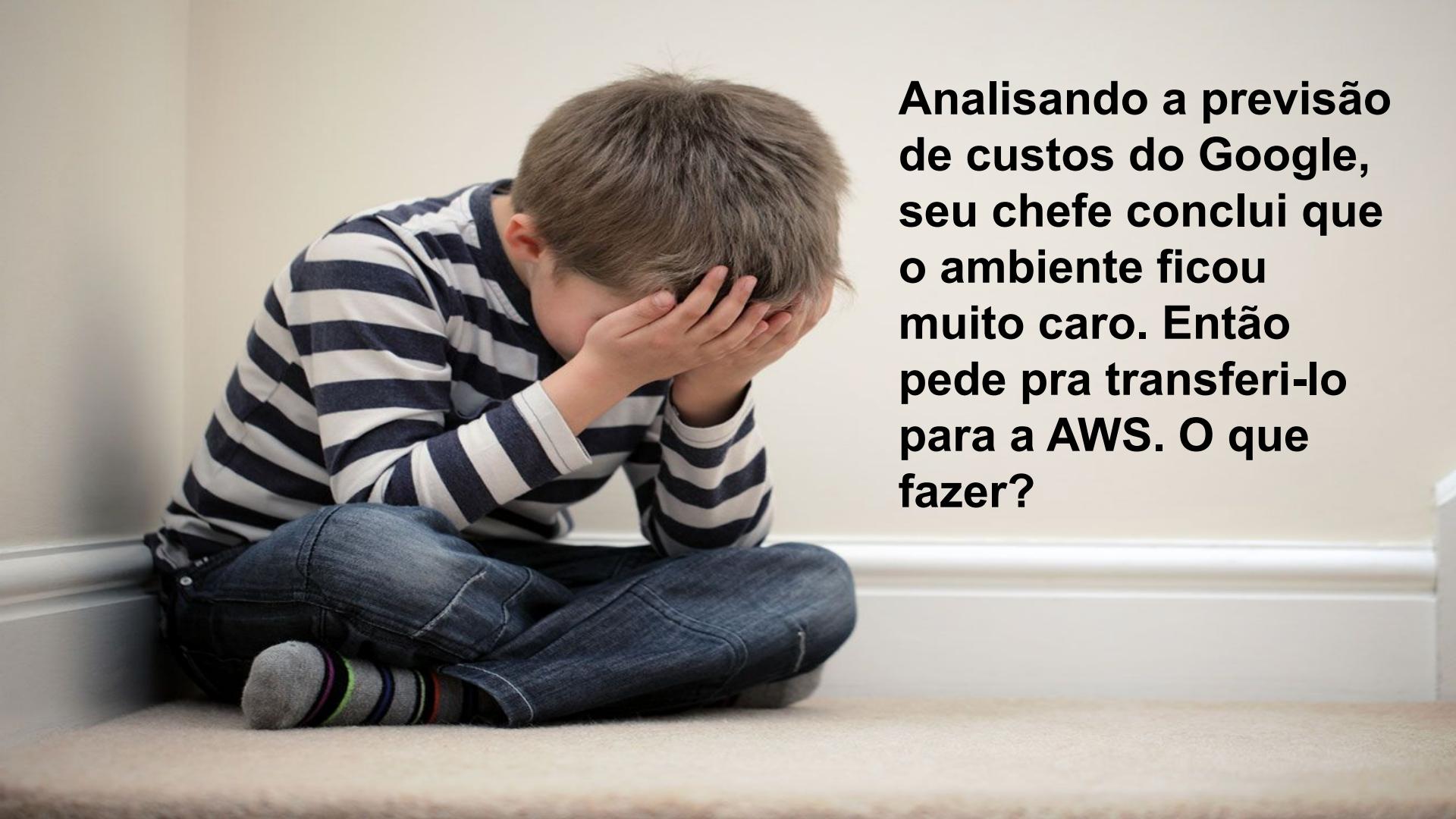
**O que você faria se o seu chefe
lhe pedisse para criar um servidor
Debian na Google Cloud?**

A close-up photograph of a young African boy with short hair, resting his chin on his hand and looking directly at the camera with a thoughtful expression.

**Agora seu
chefe resolveu
que vai querer
um Wordpress
instalado no tal
servidor, como
você o atende?**

Depois que você entrega o ambiente, seu chefe lembra de lhe avisar que o WordPress deve aguentar uma carga muito superior àquela que apenas uma instância suportaria. Portanto, o ambiente deverá ser redimensionado e reconfigurado. E agora?



A young boy with light brown hair, wearing a blue and white horizontally striped long-sleeved shirt and blue jeans, is sitting cross-legged on a light-colored carpet. He is leaning forward with his head resting in his hands, appearing to be in distress or deep thought. The background is a plain, light-colored wall.

**Analisando a previsão
de custos do Google,
seu chefe conclui que
o ambiente ficou
muito caro. Então
pede pra transferi-lo
para a AWS. O que
fazer?**

Orquestrador de Nuvem

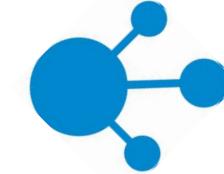


Orquestrador de Nuvem

Consiste em uma camada de software entre o provedor de nuvem e o usuário final responsável por administrar os recursos computacionais “alugados” pelo cliente junto ao provedor.



Cliente



Orquestrador



Provedores de Nuvem

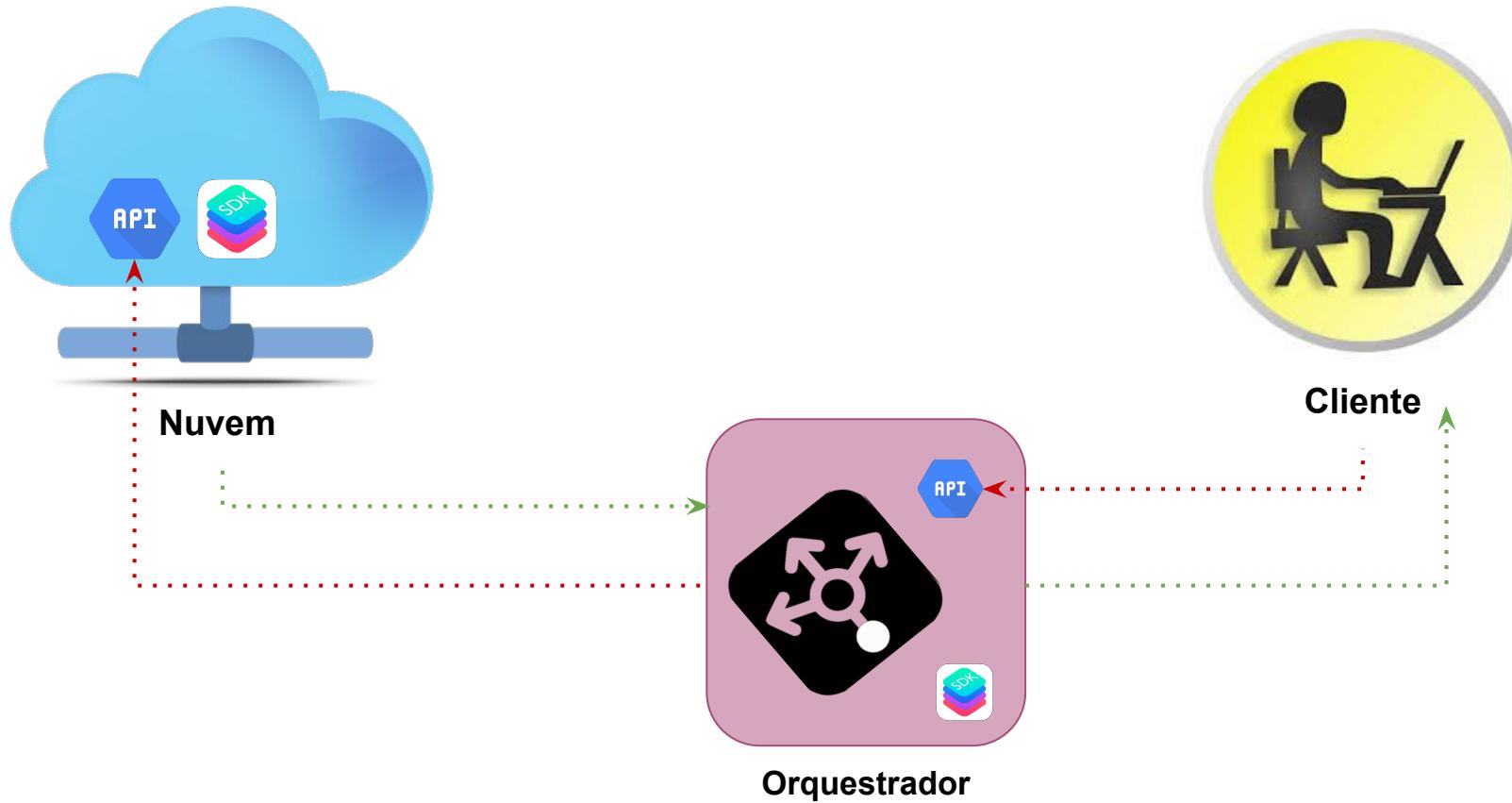


Orquestrador de Nuvem

- Aplicações
 - Provisionamento de Recursos
 - Máquinas Virtuais
 - Armazenamento
 - Plataformas
 - Etc
 - Configuração de Recursos
 - Gerenciamento da Configuração
 - Aplicar novos valores para parâmetros
 - Viabilizar utilização de recursos em diferentes provedores (caso seja multicloud)



Orquestrador de Nuvem - Funcionamento

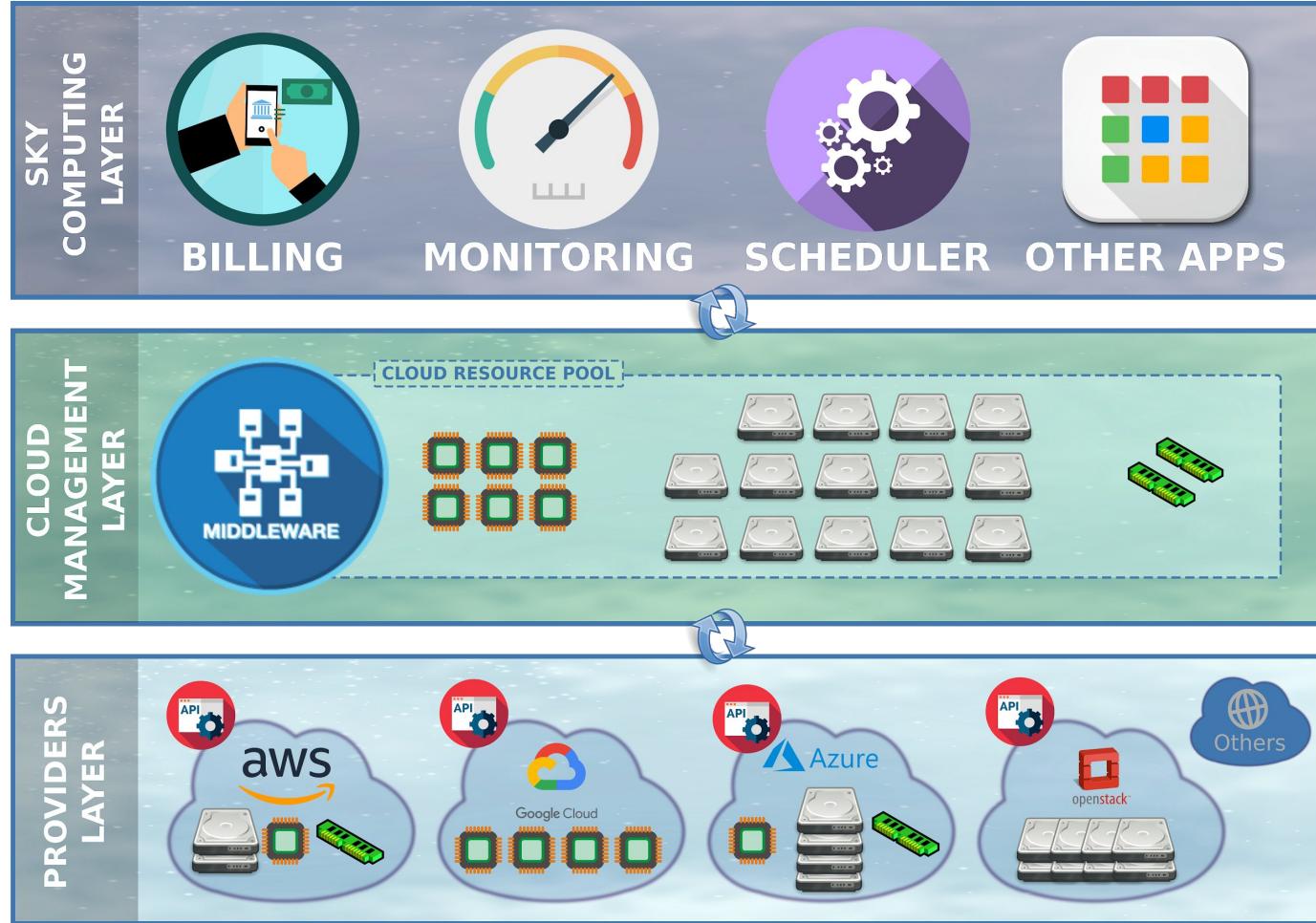




Orquestrador de Nuvem

- Desvantagens
 - Incremento de camada na solução
 - Possível atraso na inclusão de novas funcionalidades nos provedores
 - Suporte pode ser custoso (tempo e financeiro)
- Vantagens
 - Ganho de agilidade nos processos de provisionamento e desprovisionamento
 - Ambientes padronizados
 - Viabilização da utilização curta de ambiente complexos
 - Simulação de custos entre provedores
 - Testes
 - Etc
 - Possibilidade de versionar a infraestrutura
 - Pode ser utilizado em estratégias de DevOps, GitOps ou até NoOps
 - Aderente ao paradigma de Sky Computing

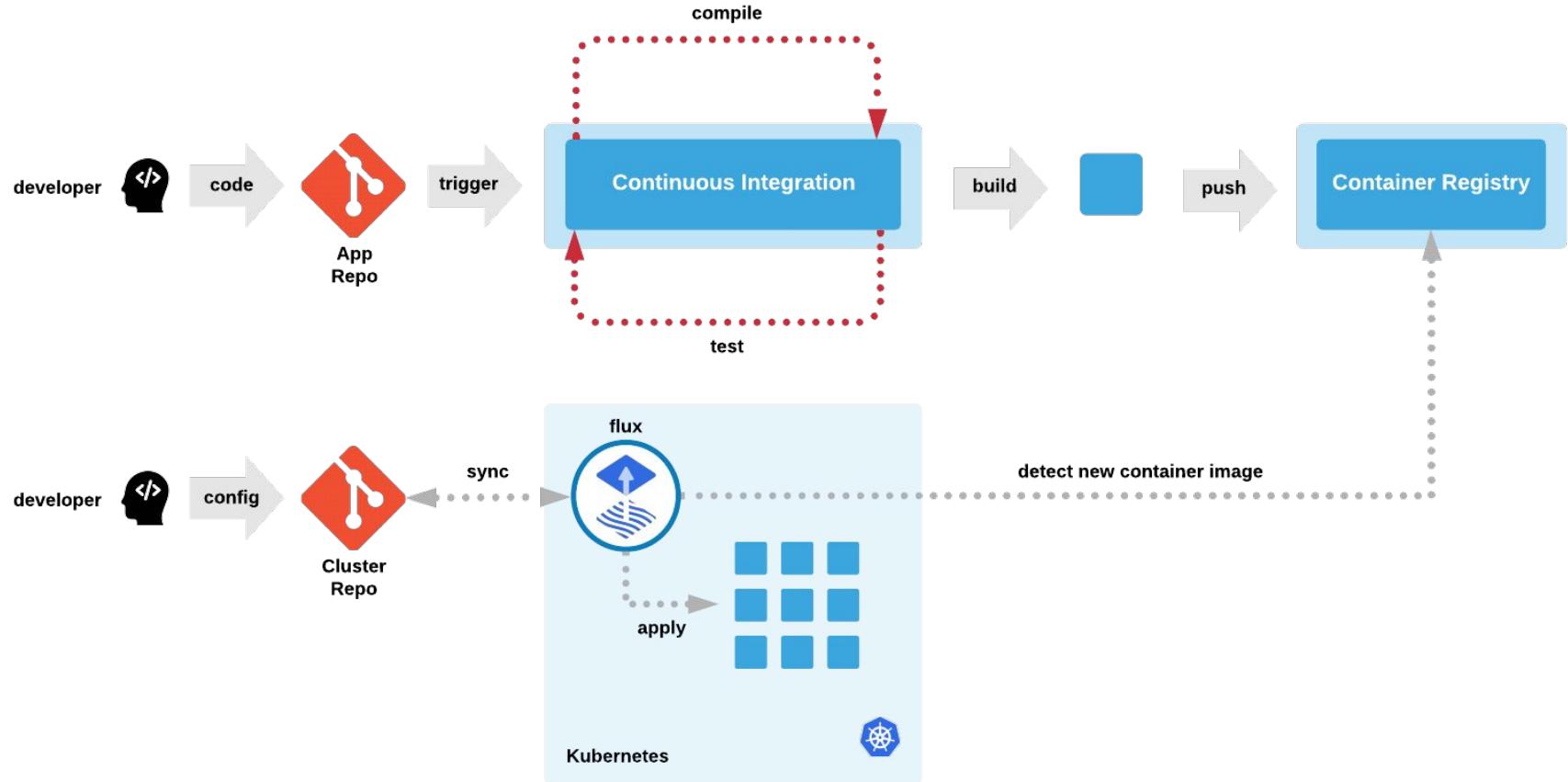
Sky Computing





Variantes:
DevSecOps / BizDevOps

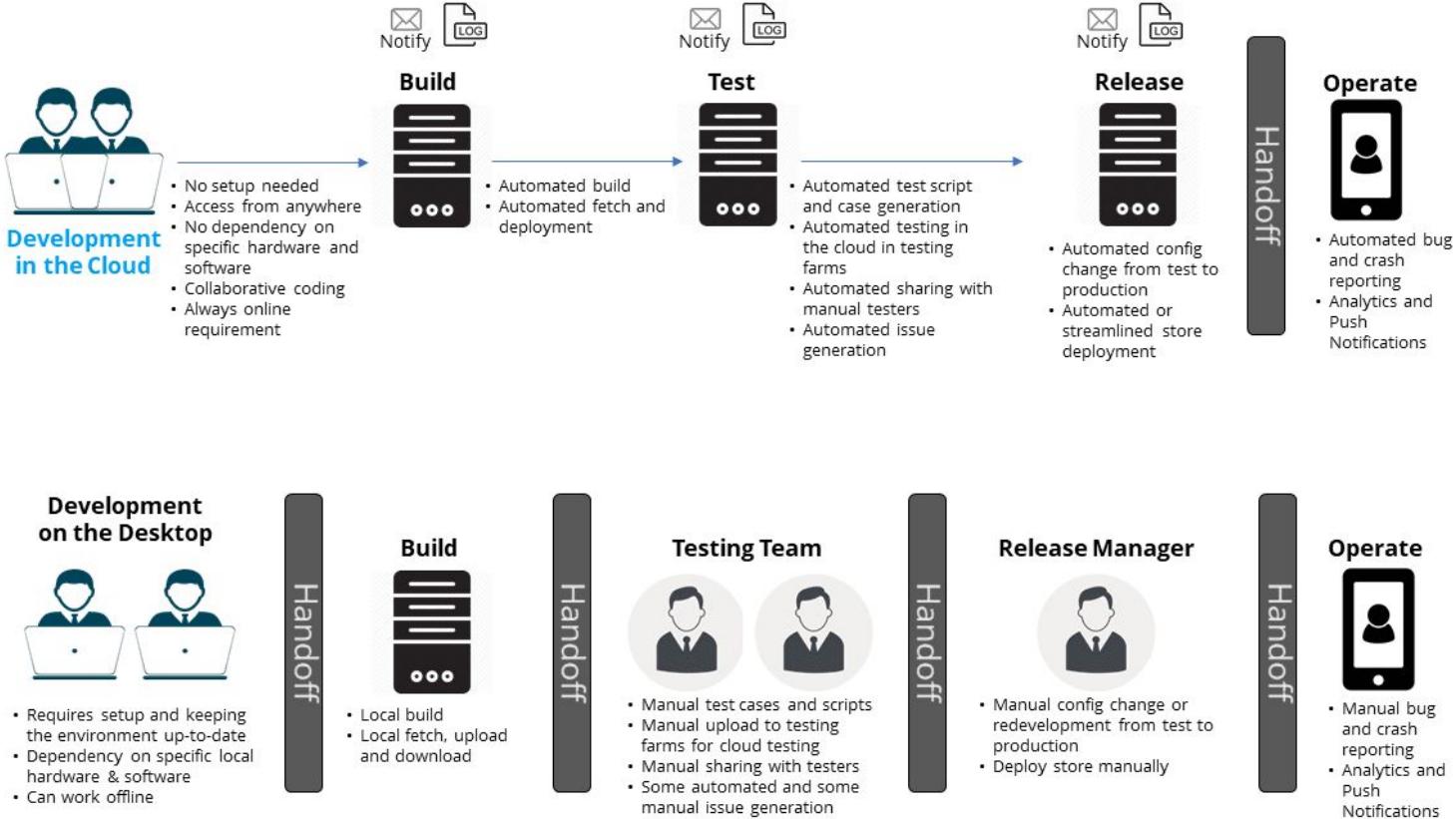
GitOps





NoOps

DevOps vs NoOps

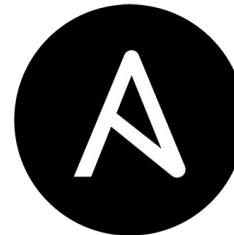


Ferramentas de Automação

Orquestradores de Nuvem são ferramentas de automação, mas nem toda ferramenta de automação é um orquestrador de nuvem. Não confundir!



SALTSTACK



ANSIBLE



Principais Orquestradores



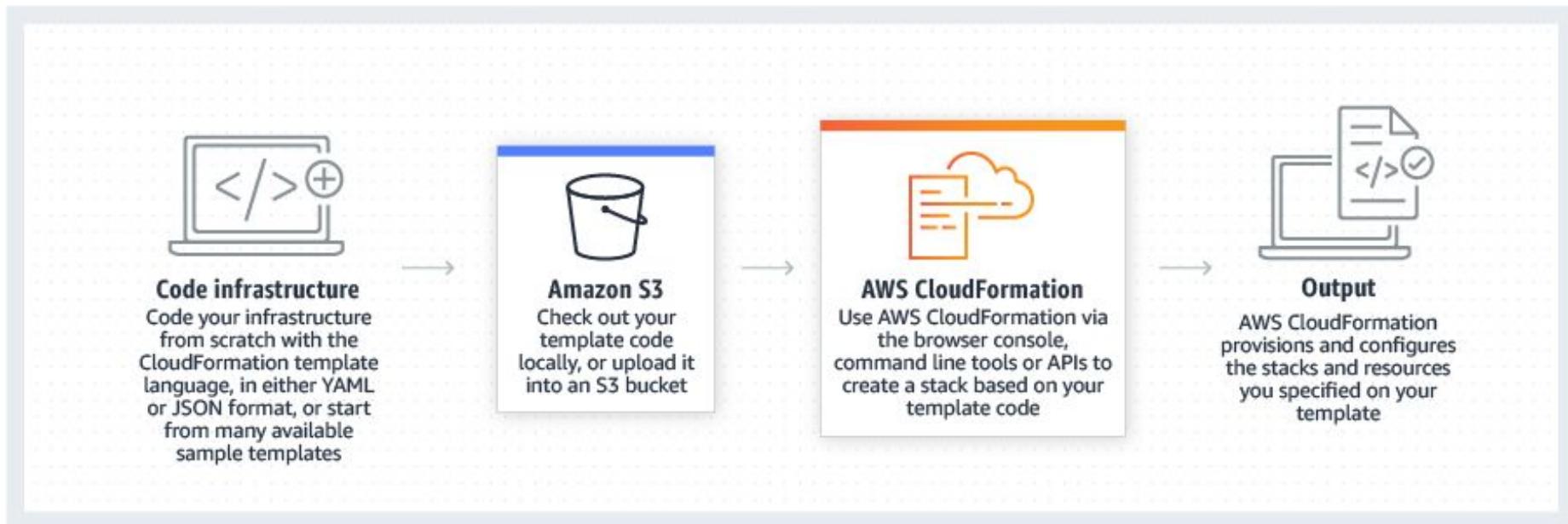
AWS
CloudFormation





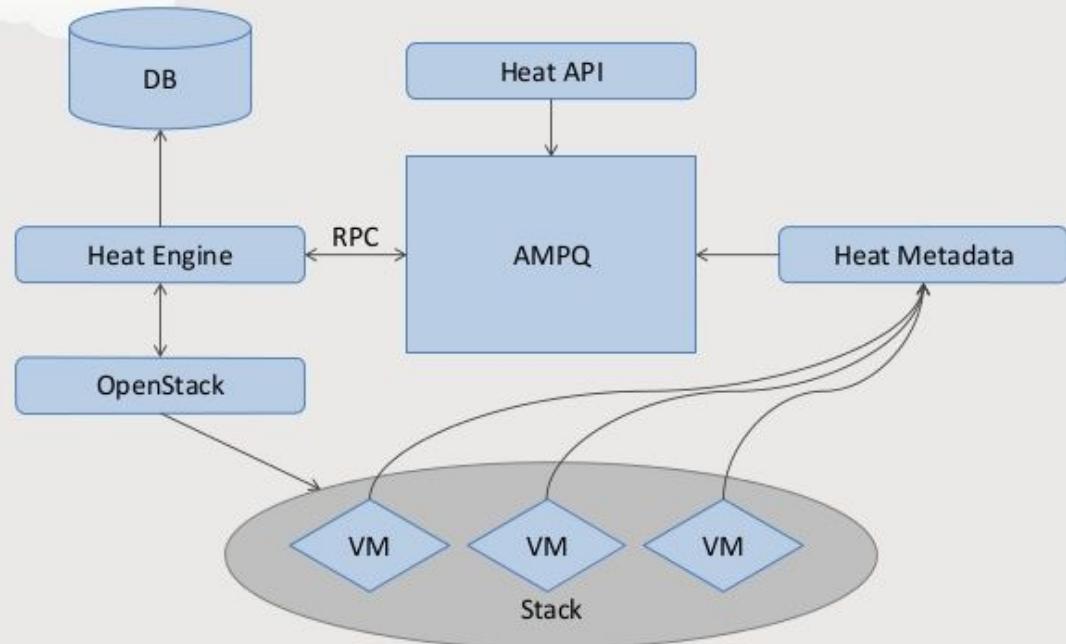
AWS
CloudFormation

- Faz parte da suíte da AWS
- Destinado a orquestrar os produtos da Amazon
- Artefato escrito em **JSON/YAML**



Heat basic architecture

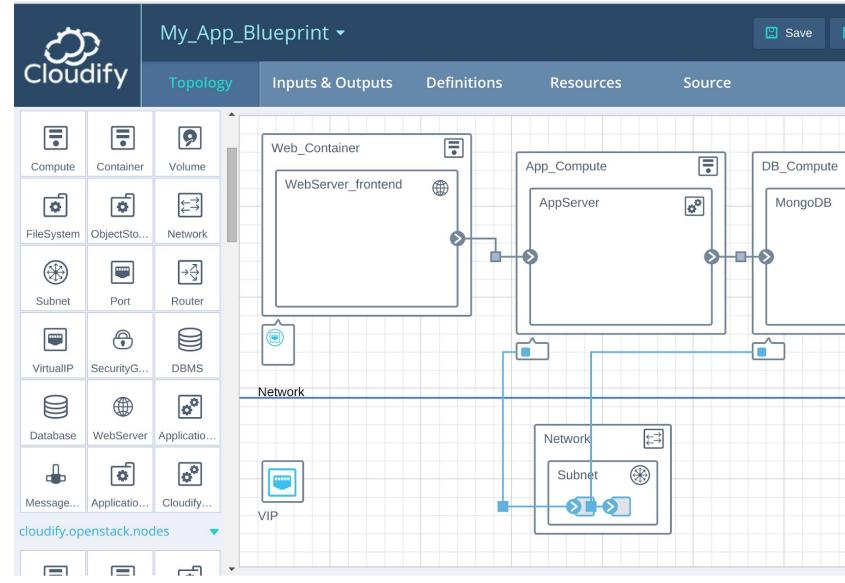
- Integrante da solução do OpenStack
- Orquestra recursos de ambientes OpenStack e permite integrações com outras soluções como o Cloud Formations



openstack.[®]



- Baseado na arquitetura denominada TOSCA
- Gerenciamento gráfico
- Permite que os usuários gerenciem diferentes domínios de orquestração e automação como parte de um pipeline comum de CI / CD
- Modelagem de serviços compostos, contendo componentes de vários serviços Cloudify e outros domínios de orquestração
- Aceita plugins





- Linguagem própria e simples
- Leve
- Mantém estado do ambiente
- Documentação robusta
- Suporta diversos provedores e plataformas
- Aceita plugins
- Interface CLI apenas

Vamos praticar?





Pré-requisitos

- Conta ativa no Google Cloud Platform
- Existência de uma conta de serviço com as seguintes permissões:
 - Administrador do Compute
 - Ativar Compute API
 - https://console.developers.google.com/apis/api/compute.googleapis.com/overview?project=id_do_projeto_do_aluno*
 - Usuário da conta de serviço
 - Administrador do Cloud SQL
 - Ativar Cloud SQL API
 - https://console.developers.google.com/apis/api/sqladmin.googleapis.com/overview?project=id_do_projeto_do_aluno*
- Arquivo .json contendo as credenciais de acesso

* Encontre a id do projeto no arquivo json obtido junto ao provedor



Objetivo

Provisionar ambientes no Google Cloud Platform por meio do Terraform usando diferentes abordagens:

1. Criar o primeiro blueprint
2. Provisionar uma instância no GCP usando Terraform
3. Provisionar uma instância com Apache instalado
4. Provisionar uma instância com Apache acessível
5. Provisionar uma instância contendo uma aplicação Node.js + MySQL
6. Provisionar um ambiente contendo 3 instâncias da aplicação
7. Provisionar um ambiente que utilize balanceador de carga



Baixe o Terraform

Acesse: <https://www.terraform.io/downloads.html>

Baixe o executável do terraform.

Mova o executável do terraform para o diretório **/usr/sbin** de modo que ele possa ser executado em qualquer ponto do Sistema Operacional

Atividade 1/7





Criando seu primeiro blueprint

Use como referência o blueprint:

<https://tinyurl.com/y5vmz2bf>

Crie seu primeiro blueprint e teste-o com:

- **terraform init**
- **terraform plan**
- **terraform apply**
- **terraform --help**

Atividade 2/7





Criando uma instância no GCP

Pesquise na documentação oficial do terraform qual recurso deve ser usado para provisionar uma instância de máquina virtual junto ao GCP e acrescente esse recurso ao seu blueprint.

Blueprint de referência:

<https://tinyurl.com/y4lmpygf>

Atividade 3/7





Criando uma instância no GCP com Apache

Configure o script de start da instância para instalar o Apache.

Blueprint de referência:

<https://tinyurl.com/y3dmrq67>



Verifique se a sua instância está acessível

Teste, usando um navegador, se a porta 80 está respondendo.

Você pode obter o IP a partir da console do provedor ou a partir da saída do Terraform.



Atividade 4/7



Garanta que a sua instância esteja acessível

Altere seu blueprint para permitir o tráfego HTTP até a sua instância.

Blueprint de referência:

<https://tinyurl.com/y4tvhgta>



Atividade 5/7



Provisionar uma instância com uma App

Crie um novo blueprint em outro diretório que provisione uma instância cuja configuração seja aplicada a partir de um shell script publicado no Github. Esse script deverá configurar a engine do Docker, bem como colocar em execução um contêiner MySQL 5.7 e um contêiner Node.js com uma aplicação publicada neste [repositório](#).

Blueprint de referência:

<https://tinyurl.com/y6hd9j8w>

Atividade 6/7





Aumentar a quantidade de instâncias

Altere seu novo blueprint de modo que sejam criadas 3 instâncias idênticas.

Blueprint de referência:

<https://tinyurl.com/y3ksyuh5>

Atividade 7/7





Balancear a carga do ambiente

Altere seu novo blueprint inserindo um balanceador de carga para as 3 instâncias.

Blueprint de referência:

<https://tinyurl.com/y2kfu56q>



Outras atividades interessantes

- Substituir o contêiner MySQL por um serviço de DBaaS
- Incluir uma camada de armazenamento de arquivos
- Utilizar o serviço de Kubernetes



Atenção!!!

Não esqueça de destruir o ambiente após os estudos para não receber cobranças indesejadas.

comando:

- `terraform destroy`

Obrigado

Leonardo Rebouças de Carvalho

leouesb@gmail.com