

# Lab. Nacional

# Multiusuário Testbeds -

# Cluster Nacional

- [Conceitos básicos de Kubernetes](#)
- [Conhecendo o ambiente de experimentação - Cluster Nacional](#)

# Conceitos básicos de Kubernetes



## O que é o Kubernetes?

- Sistema de código aberto.

Ele foi criado pelo Google e agora é mantido pela Cloud Native Computing Foundation (CNCF).

- Possibilita a automação e a implantação.
- Dimensionamento e a gestão de aplicativos em containers .

## Infraestrutura de um Cluster Kubernetes

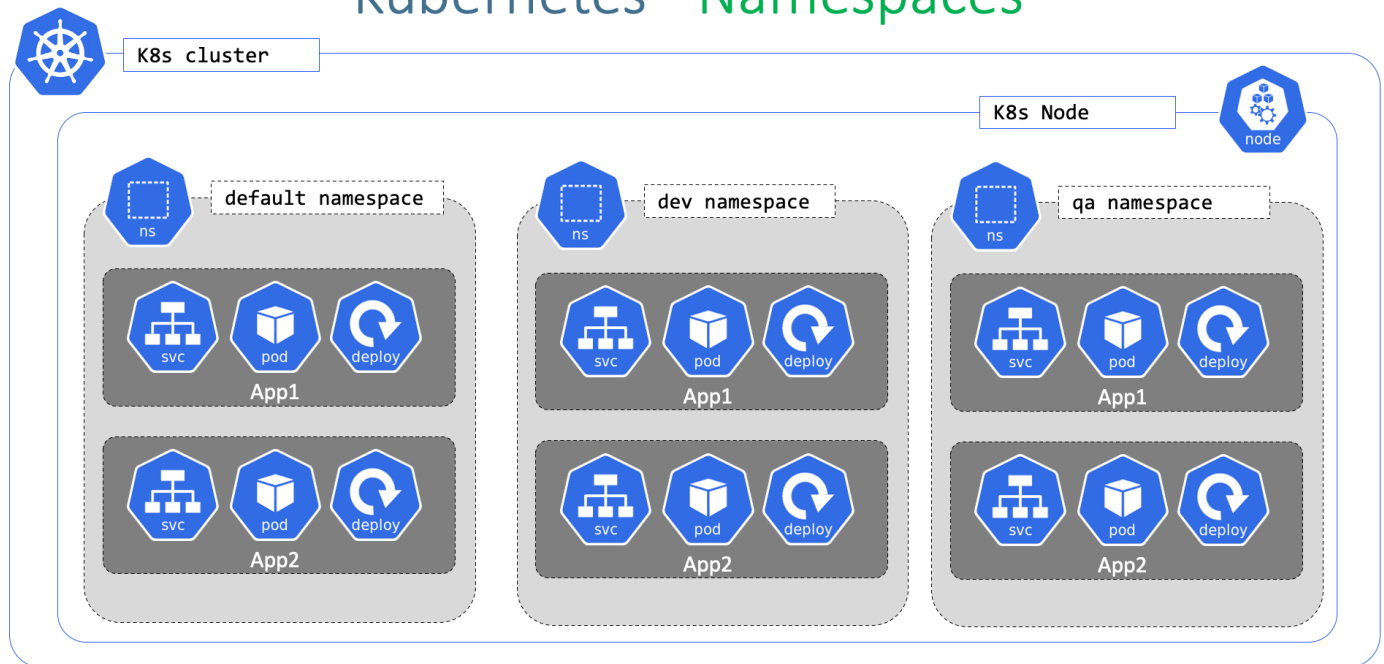
- Conjunto de servidores de processamento
- Esses servidores são chamados de Nós
- Todo cluster possui ao menos um servidor de processamento (worker node) e um Control Plane (Master).

## Descrição de alguns componentes utilizados no Kubernetes

### ***Namespaces***

- Fornecem um mecanismo para isolar grupos de recursos dentro de um único cluster
- É adicionado cotas de recursos.
- Permissões podem ser aplicadas.
- Objetos Kubernetes podem ser criados em um namespace (Deployments, Services, etc.).

# Kubernetes - Namespaces



## Pods

- Menores unidades de computação implantáveis que você pode criar e gerenciar no Kubernetes.
- Possui armazenamento compartilhado
- Recursos de rede compartilhados
- Especificação de como executar os contêineres.
- O conteúdo de um Pod é sempre executado em um contexto compartilhado.

## Deployment

- É um objeto que gerencia um conjunto replicado de pods.
- Garante que um número especificado de réplicas de um aplicativo esteja em execução em um determinado momento.
- Permite atualizações e rollbacks sem tempo de inatividade.
- Você pode pensar em um deployment como uma maneira declarativa de gerenciar pods.

## Services

- É um objeto que define um conjunto lógico de pods por meio de um seletor de rótulos ( `label selector` ).
- Define uma política de acesso aos pods.
- Permitem que um conjunto de pods seja exposto como um serviço de rede.
- Pode ser usado para descoberta de serviço e balanceamento de carga interno.

## Exemplo de Service:

## **Nodeport**

- Permite a obtenção de tráfego externo diretamente para o serviço.
- Ele abre uma porta específica em todos os Nós e qualquer tráfego enviado a essa porta é encaminhado para o serviço.

## **ReplicaSet**

- É um serviço que garante que um número especificado de réplicas de POD está sendo executado.
- É utilizado para garantir a disponibilidade de um número especificado de Pods idênticos.

## **Referências:**

[Documentação oficial Kubernetes](#)

<https://www.youtube.com/watch?v=jb8NZj9wX4A>

<https://www.youtube.com/watch?v=eV4ujv8kKC8>

<https://stacksimplify.com/azure-aks/azure-kubernetes-service-namespaces-imperative/>

<https://medium.com/tecnologia-e-afins/kubernetes-f456e59332d3>

<https://napoleon.com.br/glossario/o-que-e-kubernetes-liveness-probes/>



# Conhecendo o ambiente de experimentação - Cluster Nacional

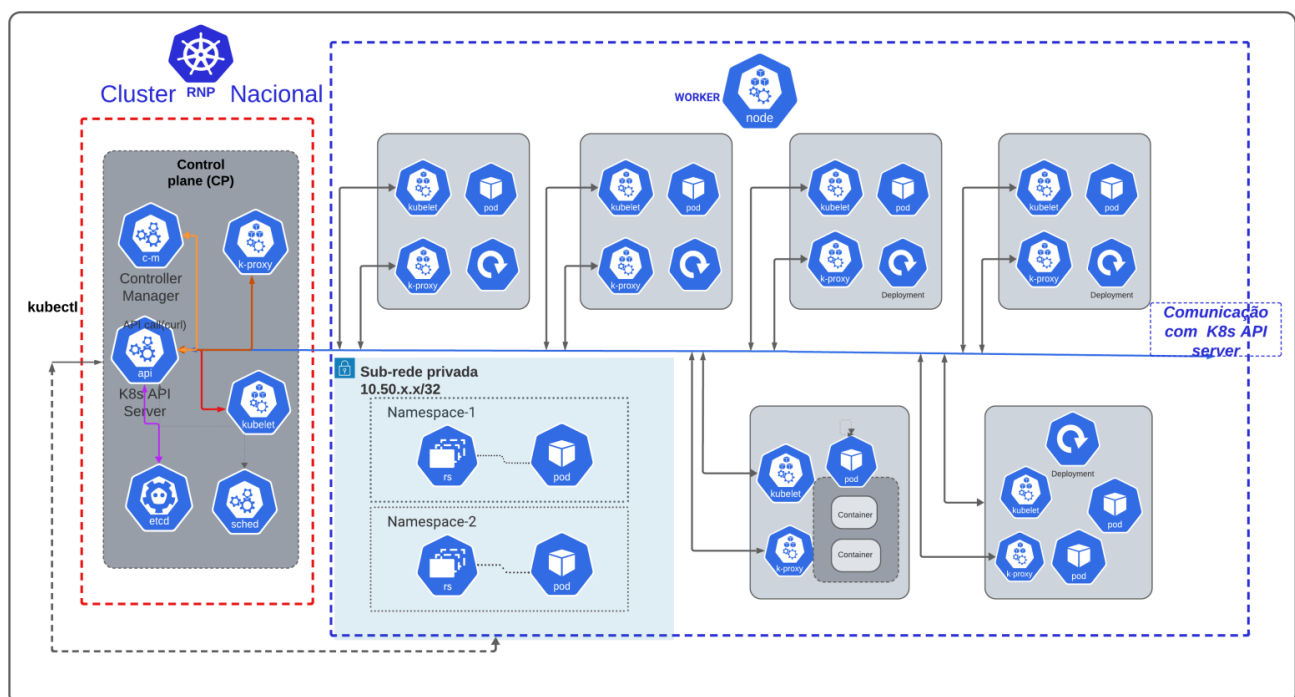


## \_1-Diagrama do Cluster Nacional

Atualmente o Cluster Nacional possui **8 nodes Workers** e **2 nodes Control plane** distribuídos geograficamente nas seguintes localidades:

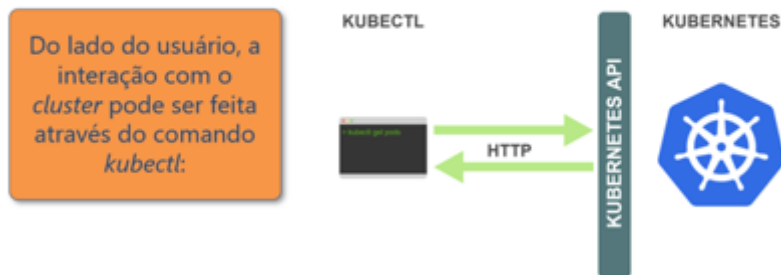
**DF,GO,PB,PE,RJ,RN,AC,MT**

- Nodes de controle e nodes de trabalho compõem a infraestrutura do serviço.
- A comunicação do nó master com os nós workers ocorre primariamente em dois sentidos,
- Primeiro é a partir do apiserver para o processo do kubelet, executado em cada nó no cluster.
- O segundo é do apiserver para qualquer nó, pod, ou serviço através da funcionalidade de proxy do apiserver.



## 2- Interagindo com o Cluster através do kubectl

O Kubernetes expõe uma API por meio do servidor API: você pode se comunicar com a API usando ***kubectl***



Veremos na prática como é feita essa interação durante o nosso hands-on.