

DOCKER + KUBERNETES



O PODER DOS CONTÊINERES NA INFRAESTRUTURA

MARCELO LINHARES E FÁBIO NITTO

DTI/MP

QUEM SOMOS?

MOTIVAÇÃO

IMPLEMENTAÇÃO / SUSTENTAÇÃO DE SISTEMAS
INFRA EM NUVEM

DIFICULDADES ENCONTRADAS

- Diversidade de fontes externas fornecendo sistemas
 - fábrica
 - desenvolvimento interno
 - demais áreas
 - fábricas contratadas pelas áreas

DIFICULDADES ENCONTRADAS

- Ausência de Documentação (Implant/Sust)
- Ausência de Padrões
- Ausência de Configurações Apropriadas
- Custo de sustentação (recursos humanos)
- Alto risco ao migrar

CONTEINERIZAÇÃO / ORQUESTRAÇÃO

- Borg - Google
- gmail, maps, search, docs, etc
- de Borg para Kubernetes

COMO ESTÁ NO MP

- começamos a estudar em 02/2016
- fase experimental (em produção)
- AFD - Assentamento Funcional Digital
- SEI - Sistema Eletrônico de Informações (desde 18/07)
- Rancher - Nuvem

SEI-MP

Total Processos:	350 Mil
Total de Documentos:	+2 Milhões

Banco de Dados:

Transações	300 / segundo
Crescimento:	1.4 GB / dia 11GB / dia (banco + anexos)

Balanceador:

Hits	1.1 Milhão / dia
Dados Ofertados:	85GB / dia
Dados Recebidos	35GB / dia

	SEI - ICS Serpro - VMs				
	Qtd de Nós	Processadores / Nó	Ram / Nó (GB)	Total Processadores	Total RAM (GB)
Balanceador	1	4	4	4	4
Aplicação	5	4	16	20	80
Memcached	0	0	0	0	0
Solr	1	4	16	4	16
Jod	1	4	8	4	8
NFS	1	4	4	4	4
			Total	36	112
	SEI - Kubernetes				
Balanceador	Cluster	Cluster	Cluster		
Aplicação	Cluster	Cluster	Cluster		
Memcached	Cluster	Cluster	Cluster		
Solr	Cluster	Cluster	Cluster		
Jod	Cluster	Cluster	Cluster		
NFS	Storage	Storage	Storage	NA	NA
			Total		
PS.: A velocidade de um processador na Nuvem (ICS) é mais que o dobro do nosso Vmware					

	SEI - ICS Serpro - VMs					
	Qtd de Nós	Processadores / Nós	Ram / Nó (GB)	Total Processadores	Total RAM (GB)	
Balanceador	1	4	4	4	4	4
Aplicação	5	4	16	20	80	
Memcached	0	0	0	0	0	0
Solr	1	4	16	4	16	
Jod	1	4	8	4	8	
NFS	1	4	4	4	4	
			Total	36	112	
	SEI - Kubernetes					
Balanceador	Cluster	Cluster	Cluster	1	0.1	
Aplicação	Cluster	Cluster	Cluster	21	33	
Memcached	Cluster	Cluster	Cluster	0.1	0.1	
Solr	Cluster	Cluster	Cluster	3	8	
Jod	Cluster	Cluster	Cluster	0.3	0.5	
NFS	Storage	Storage	Storage	NA	NA	
			Total	25.4	41.7	

O QUE É CONTÊINER?

- Conceito similar ao de Máquina Virtual.
- Virtualização a nível de SO.

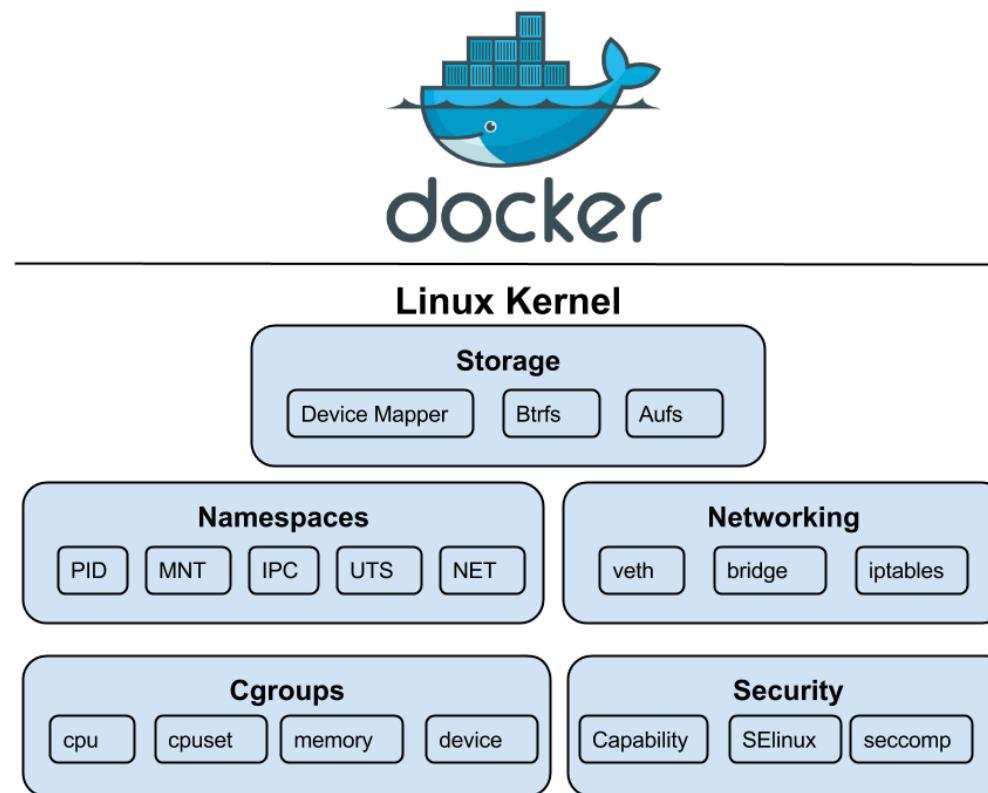
Formalmente, é um processo ou grupo de processos que executam de forma isolada:

- Processo (PID)
- Usuários
- Rede
- Recursos (CPU, RAM, Disco)
- Estrutura de diretórios
- Bibliotecas

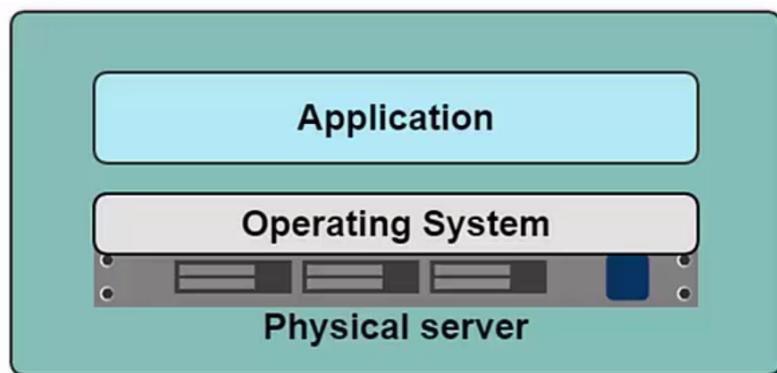
- Implementado com LXC (Linux Containers - 2008)
- Interface direto com o Kernel do linux:
 - Namespaces
 - Cgroups
 - Capabilities
 - APParmor e SELinux
 - Seccomp Policies
 - Chroots

**INTERFACE DIRETO COM O KERNEL?
PARECE DIFÍCIL...**

DOCKER SIMPLIFICA O USO:



MÁQUINA FÍSICA

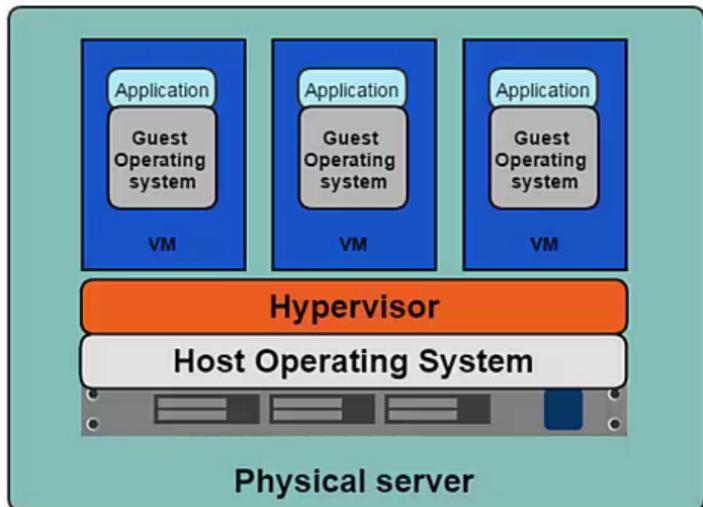


- Deployment lento
- Alto custo
- Desperdício de recursos
- Dificuldade de escalar
- Dificuldade de migração

VIRTUALIZADORA

GANHOS

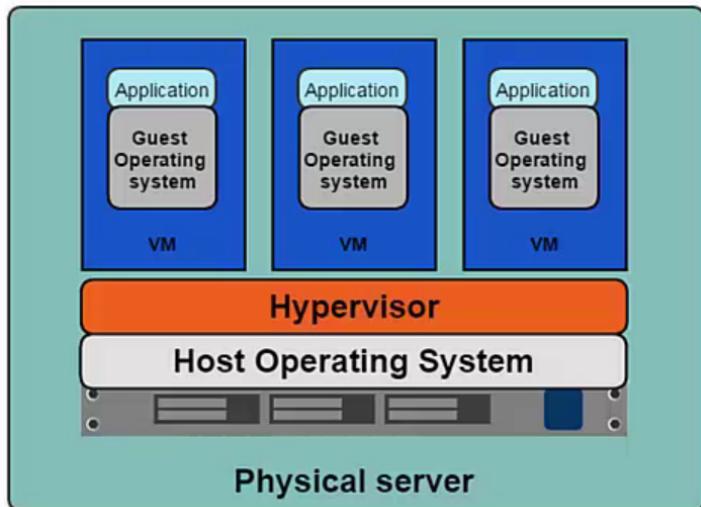
- Pool de recursos
- Maior facilidade de escala
- VMs na nuvem
 - Elasticidade
 - Pay as you go



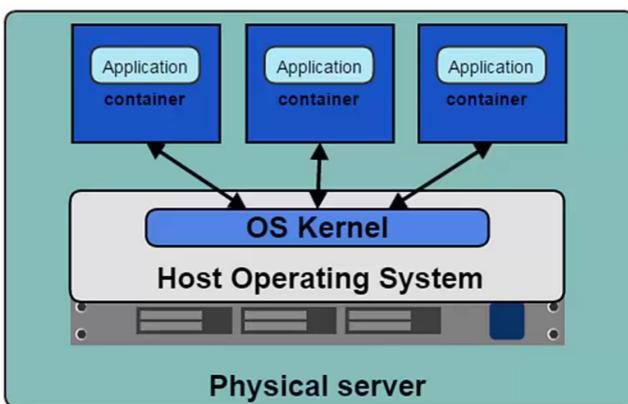
VIRTUALIZADORA

DIFICULDADES

- Alocação de recursos ainda necessária
- SO guest necessário
 - recursos desperdiçados
- Portabilidade não garantida

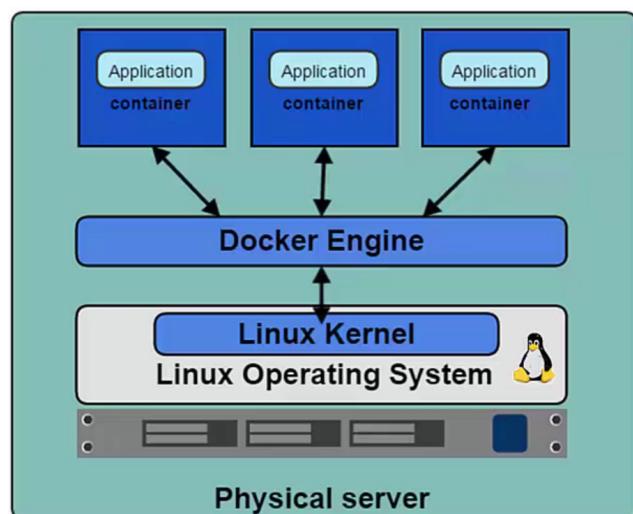


CONTÊINERES



- Mais leves (tamanho)
- Sem SO guest necessário
- Menos recursos necessários (CPU, RAM, Storage)
 - Mais containers por host

DOCKER ENGINE



- API
- Empacotamento
- DevOps (Separação de Funções)
- Densidade
- Maior portabilidade

CONCEITOS BÁSICOS DE DOCKER

CONCEITOS:

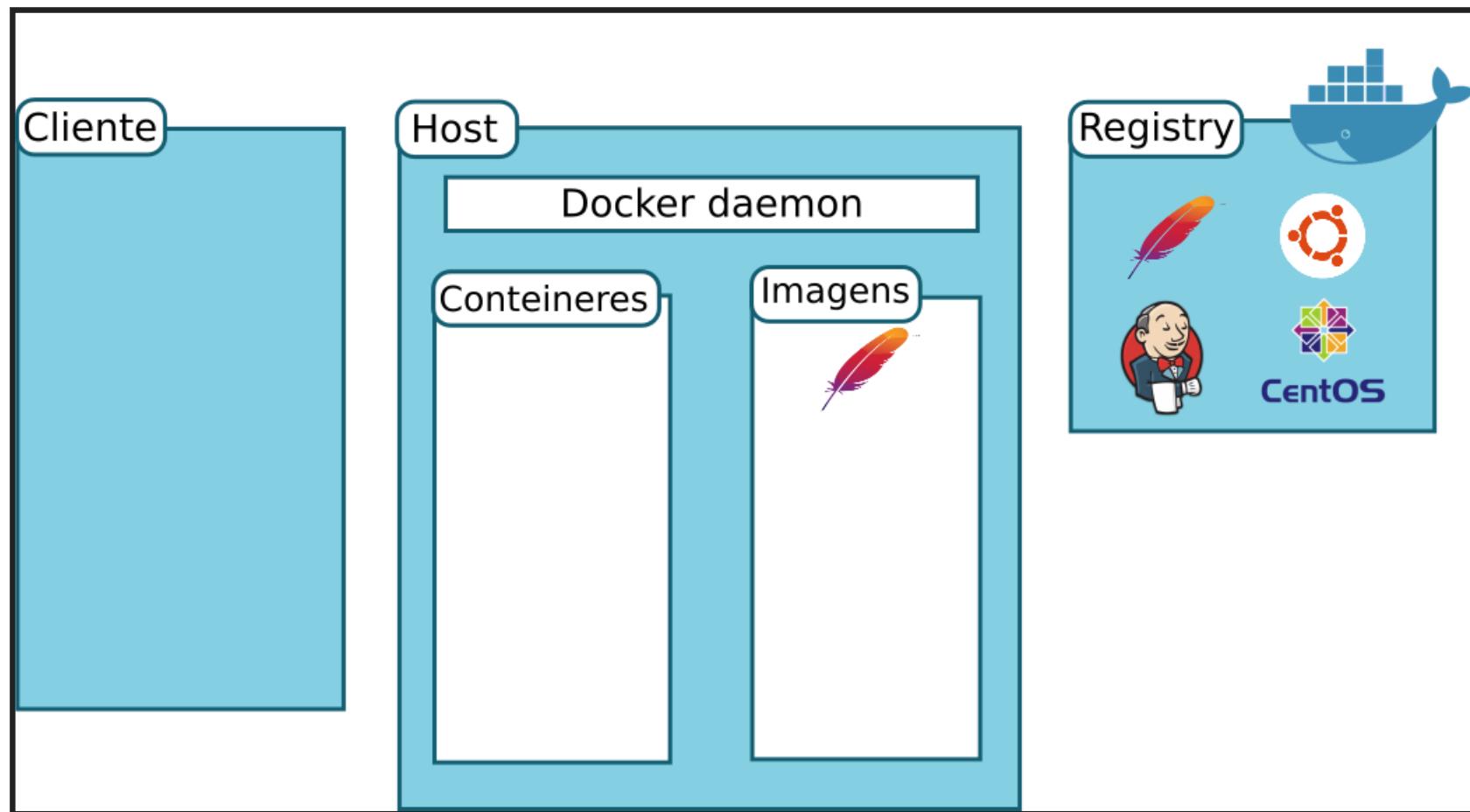
- Conteiner
- Imagem
- Registry

CONCEITOS BÁSICOS DE DOCKER

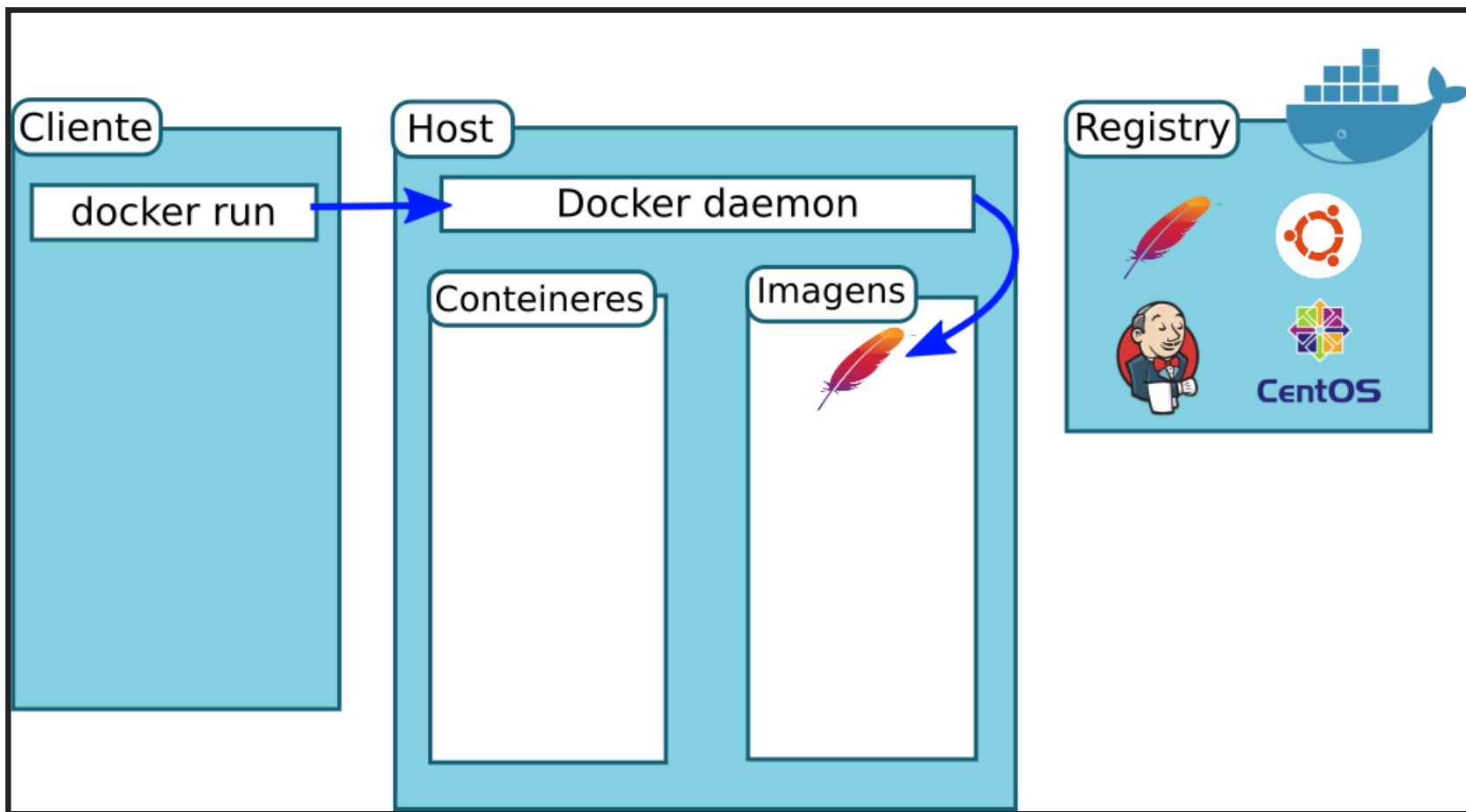
COMANDOS:

```
$ docker run  
$ docker pull  
$ docker build  
$ docker stop
```

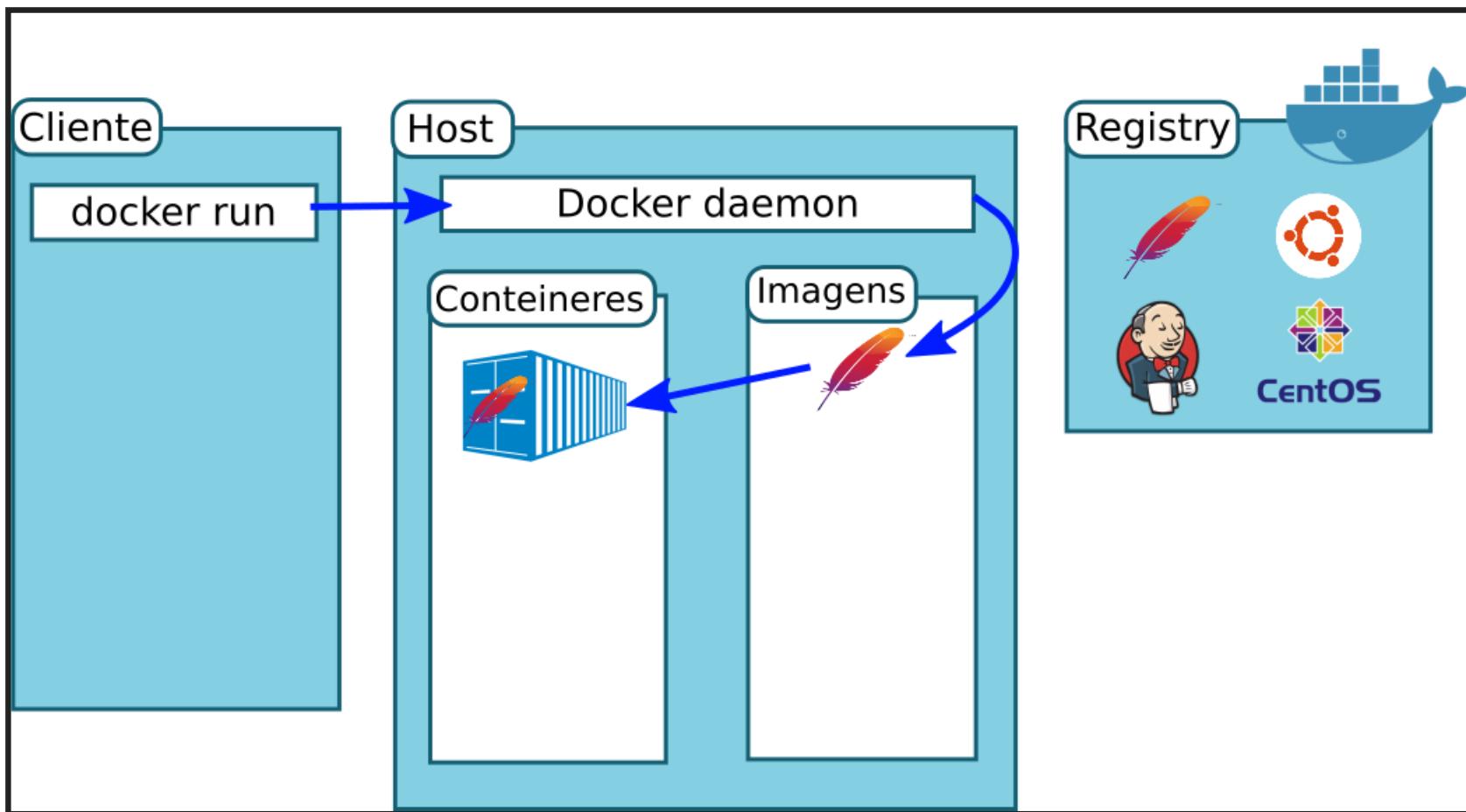
COMPONENTES DOCKER



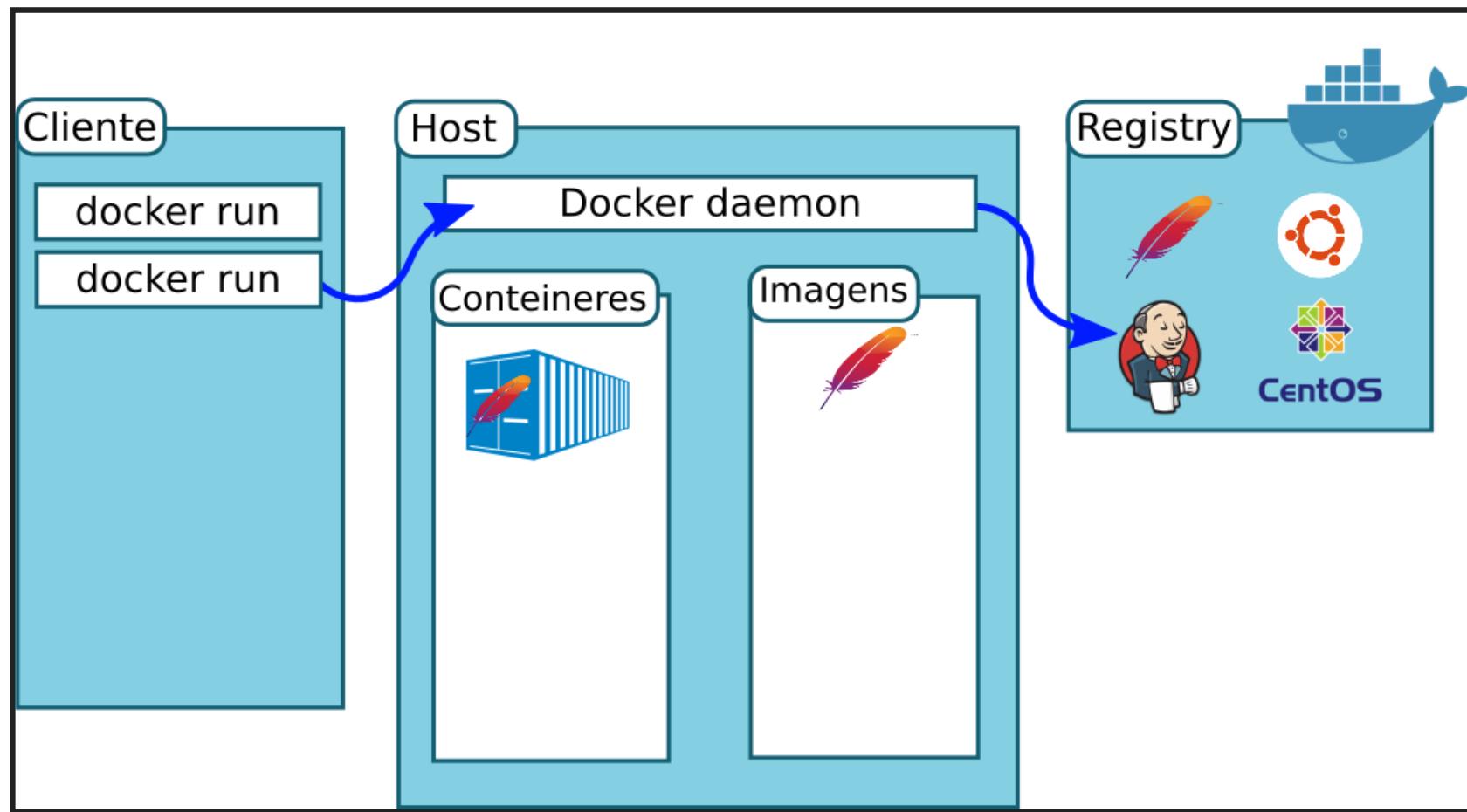
DOCKER RUN



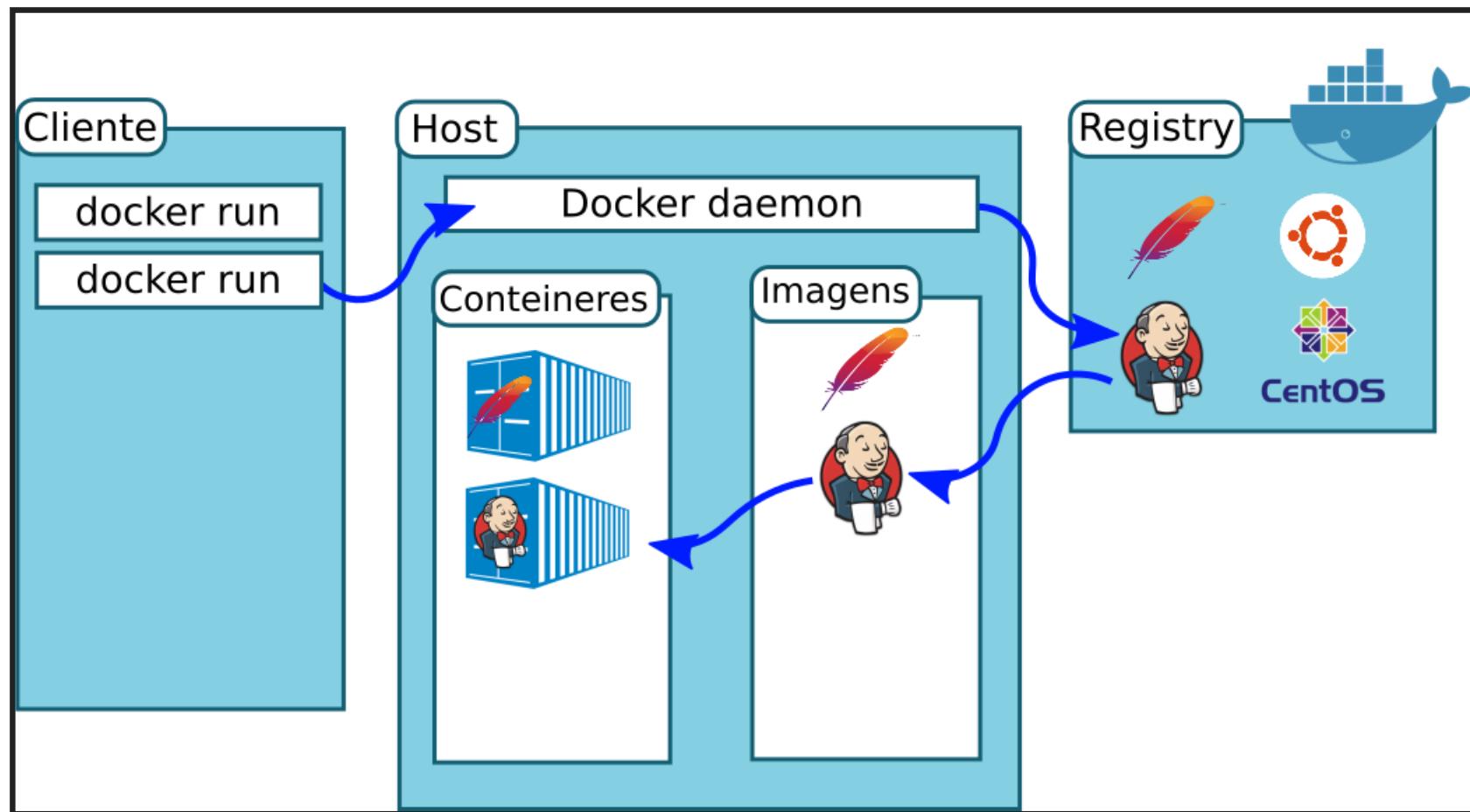
DOCKER RUN



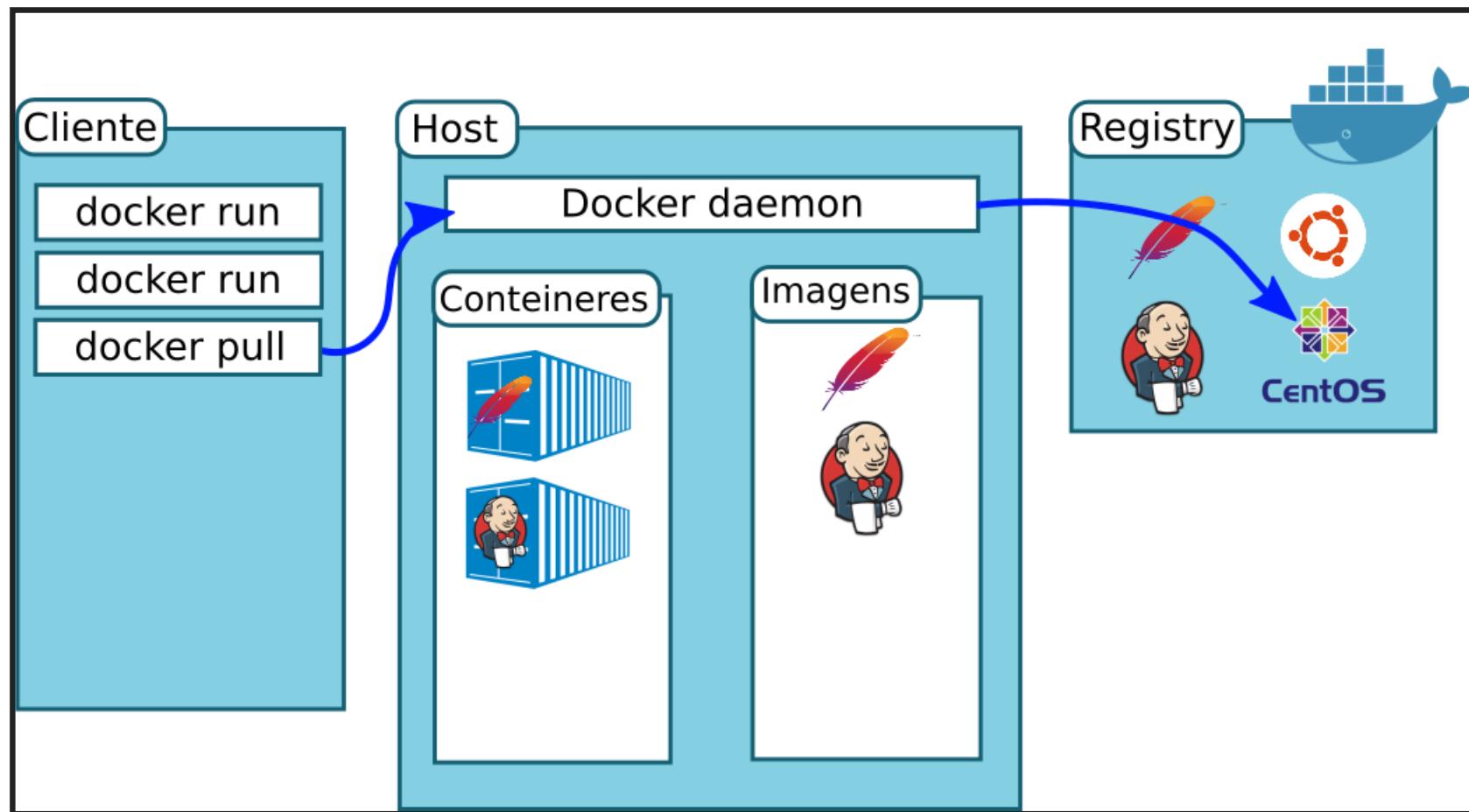
DOCKER RUN



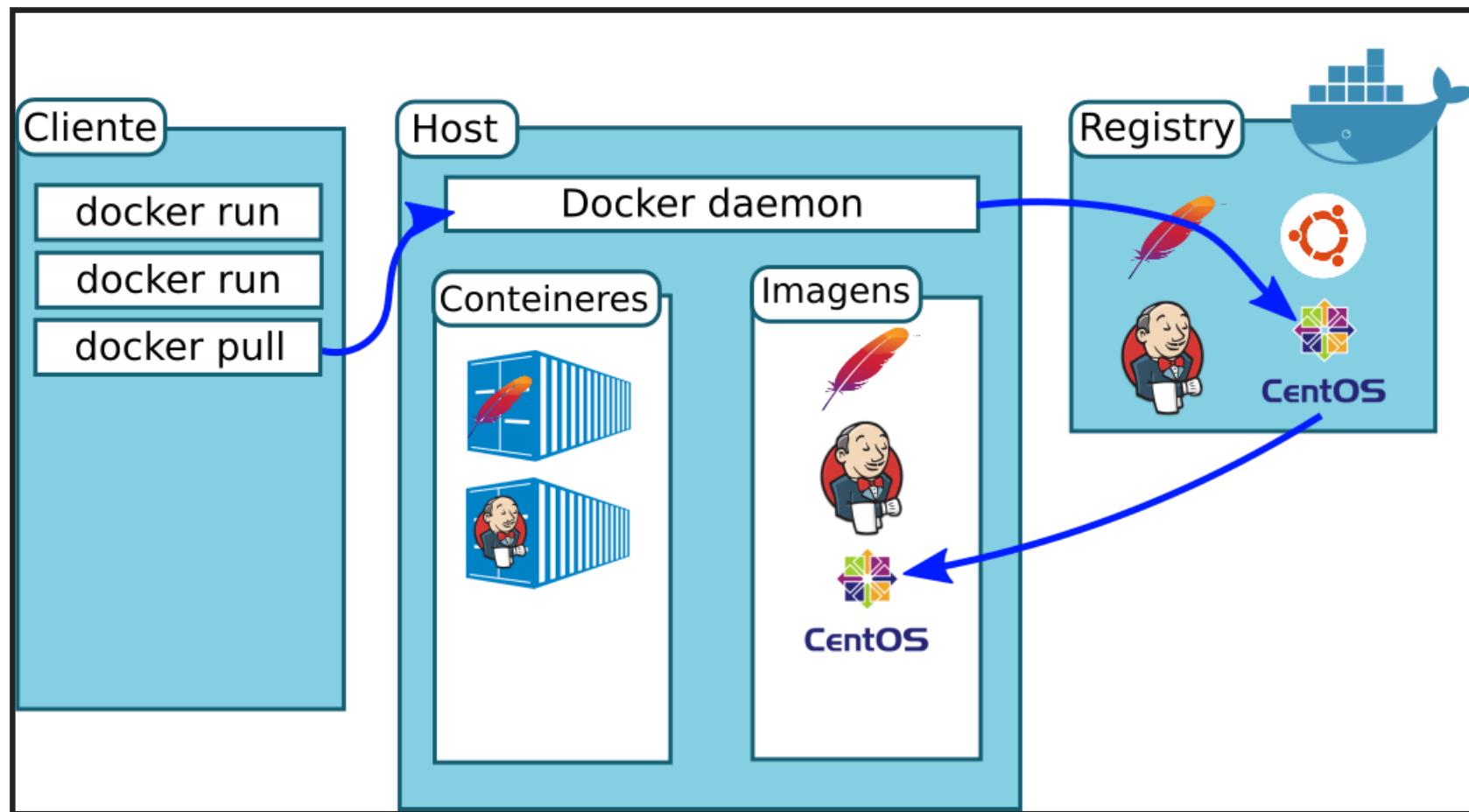
DOCKER RUN



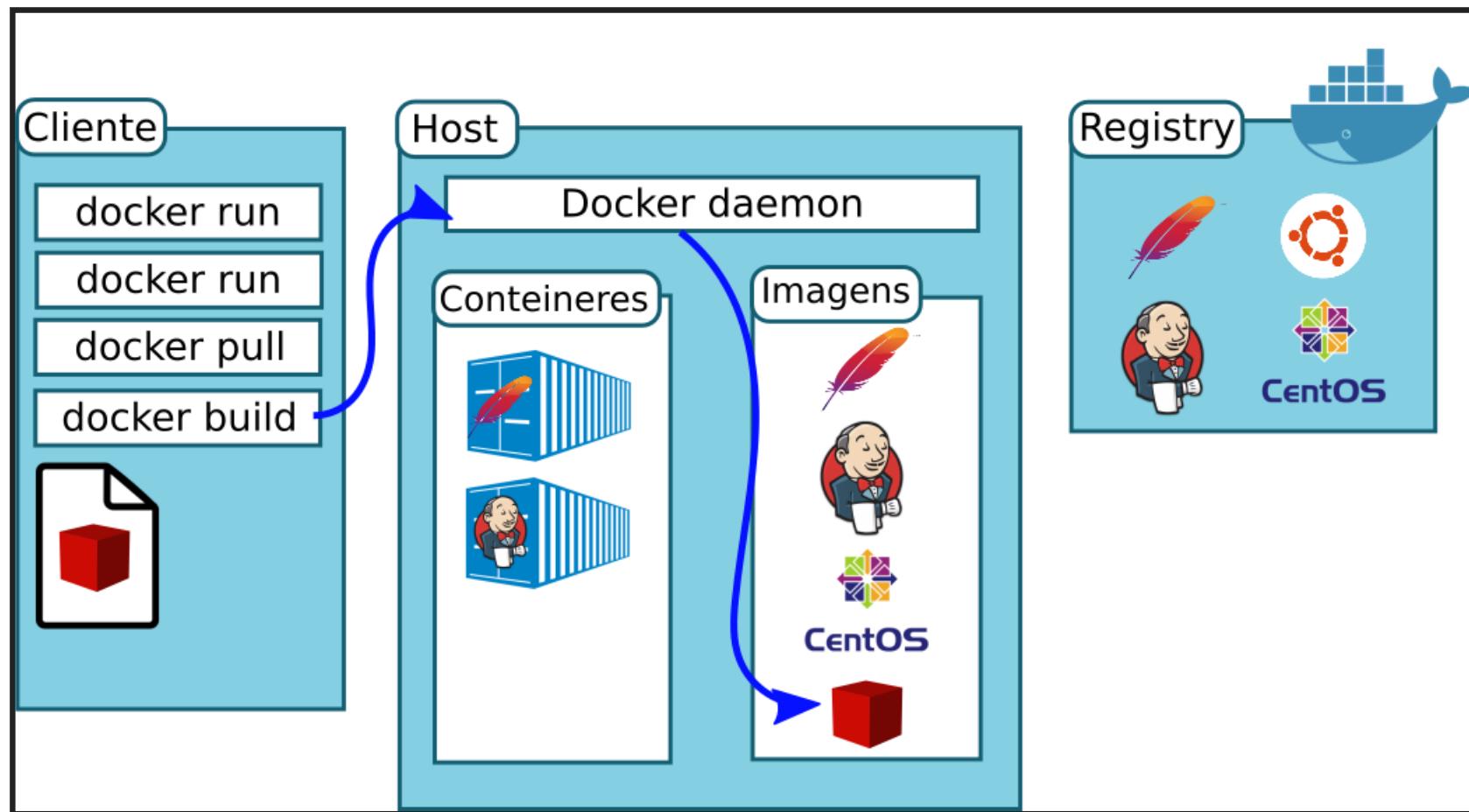
DOCKER PULL



DOCKER PULL



DOCKER BUILD

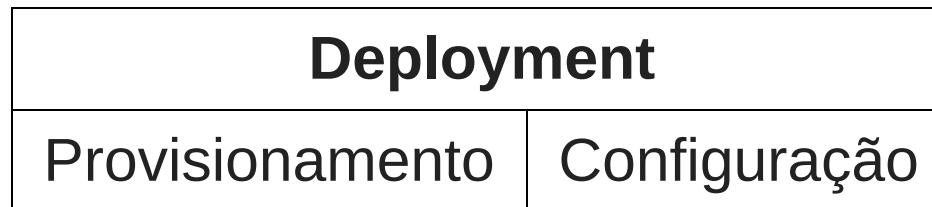


Demo

PROVISIONAMENTO, DEPLOYMENT E CONFIGURAÇÃO

2 Visões:

1.



2.



- **Provisionamento:** Preparar a maquina/host/cluster com sistema e bibliotecas
- **Deployment:** Entregar a aplicação
- **Configuração:** Configurar a aplicação

Ferramentas como:

- Puppet
- Chef
- Ansible
- Salt

Atuam, de formas diversas, em toda a linha de produção, buscando a automatização das tarefas de provisionamento, deployment e configuração.

Docker não as substituem, mas as complementam.

DOCKER COMPOSE

- Ambientes multi-conteiner
- Docker-Compose.yml
- Deploy mais fácil:
 - docker-compose up
 - docker-compose stop

Demo

ORQUESTRAÇÃO

POR QUE CLUSTER E ORQUESTRAÇÃO?

- Docker compose em apenas um host
- Alta disponibilidade
- Resiliência
- Escalabilidade automática
- Deploy e rollback imediato e sem indisponibilidade

OPÇÕES

- Kubernetes
- Docker Swarm
- Apache Mesos
- Nomad

Application containers



Máquinas Físicas ou Virtuais
Localizações Geográficas Distintas

Application containers



Kubernetes API



Máquinas Físicas ou Virtuais
Localizações Geográficas Distintas

Application containers



Kubernetes API



Máquinas Físicas ou Virtuais
Localizações Geográficas Distintas

KUBERNETES

PODs



PODs



PODs



Jboss1
10.0.0.1:8000

PODs



Jboss1
10.0.0.1:8000



PODs



Jboss1
10.0.0.1:8000



Jboss2
10.0.0.2:8000



PODs



Jboss1
10.0.0.1:8000



Jboss2
10.0.0.2:8000



PODS



Jboss1
10.0.0.1:8000



Jboss2
10.0.0.2:8000



Postgres
10.0.0.3:5432

Services



PODs



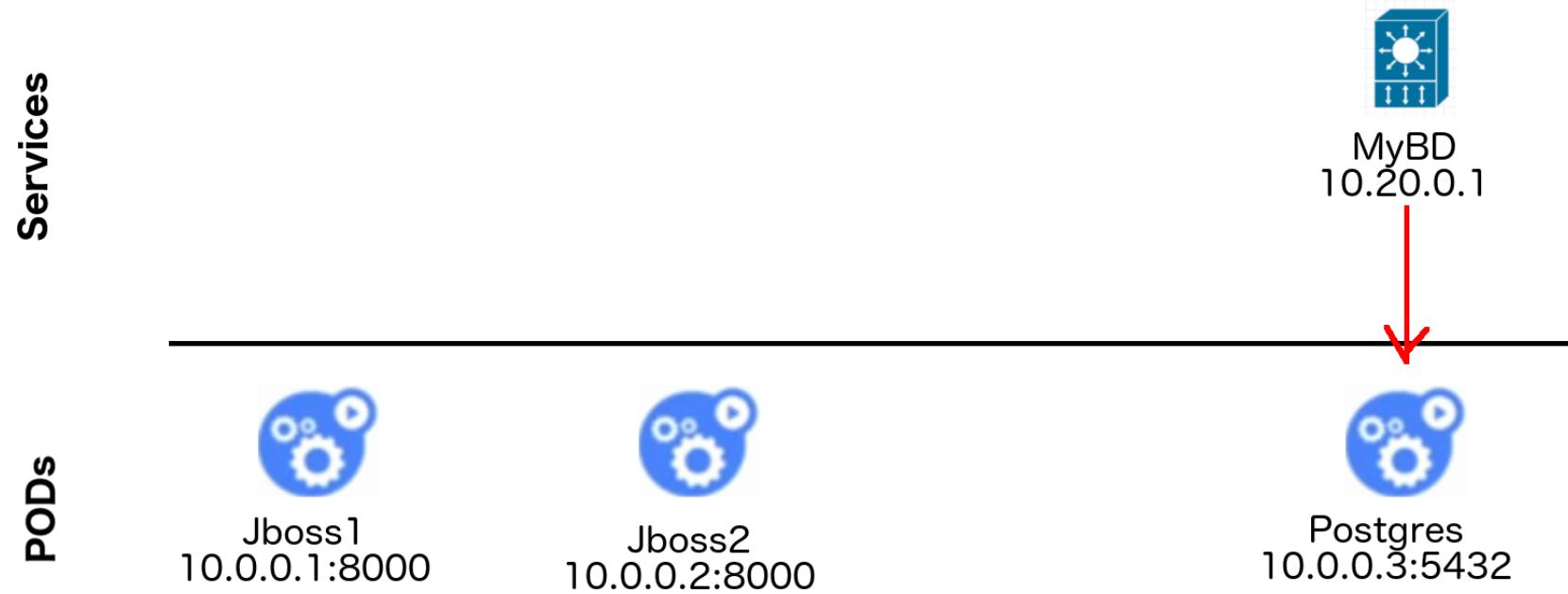
Jboss1
10.0.0.1:8000

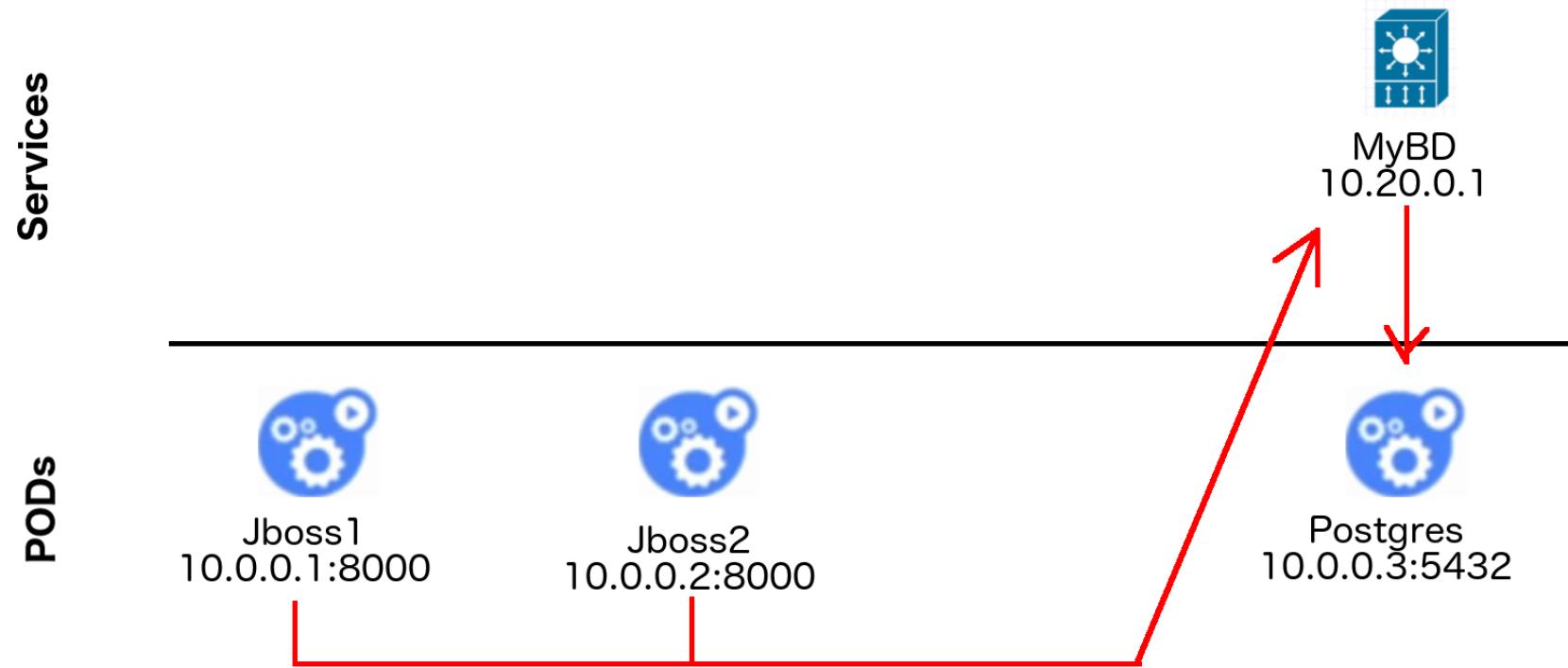


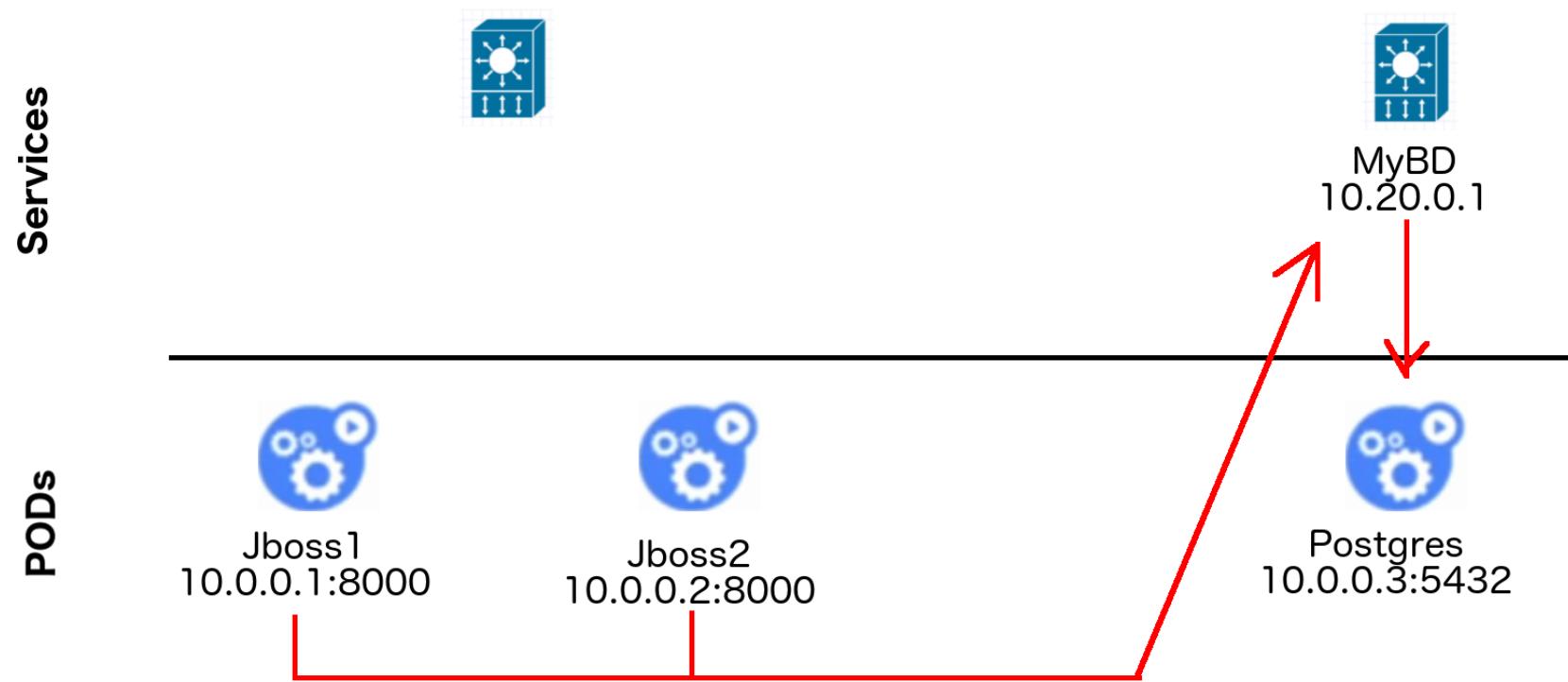
Jboss2
10.0.0.2:8000

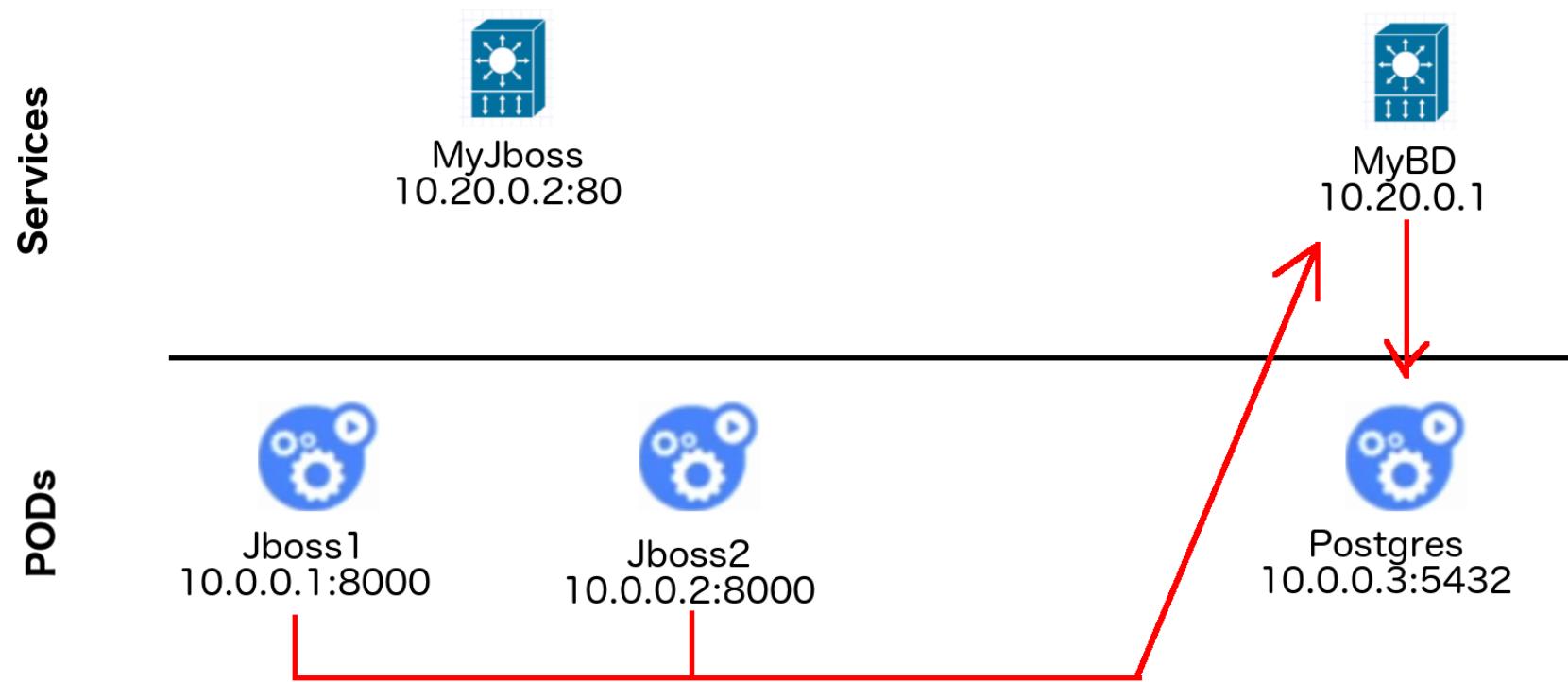


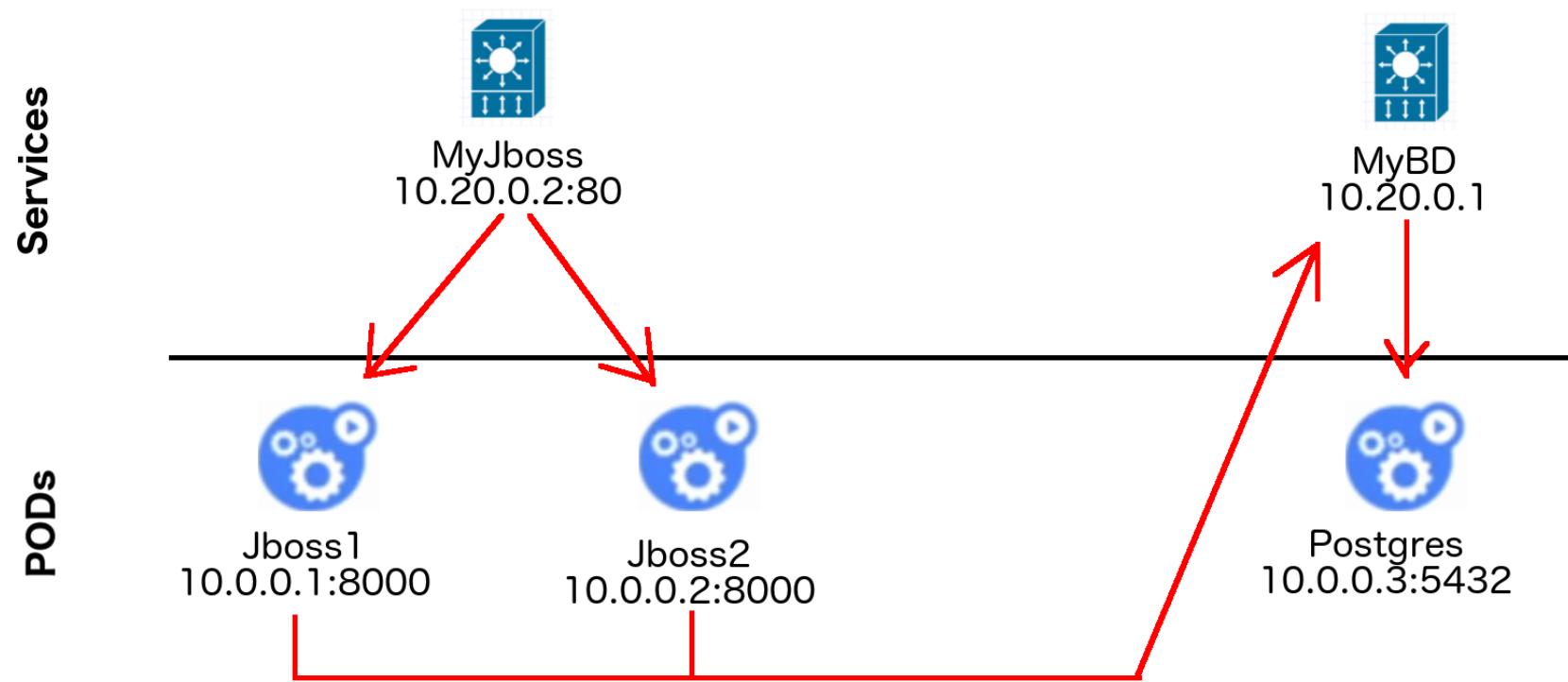
Postgres
10.0.0.3:5432

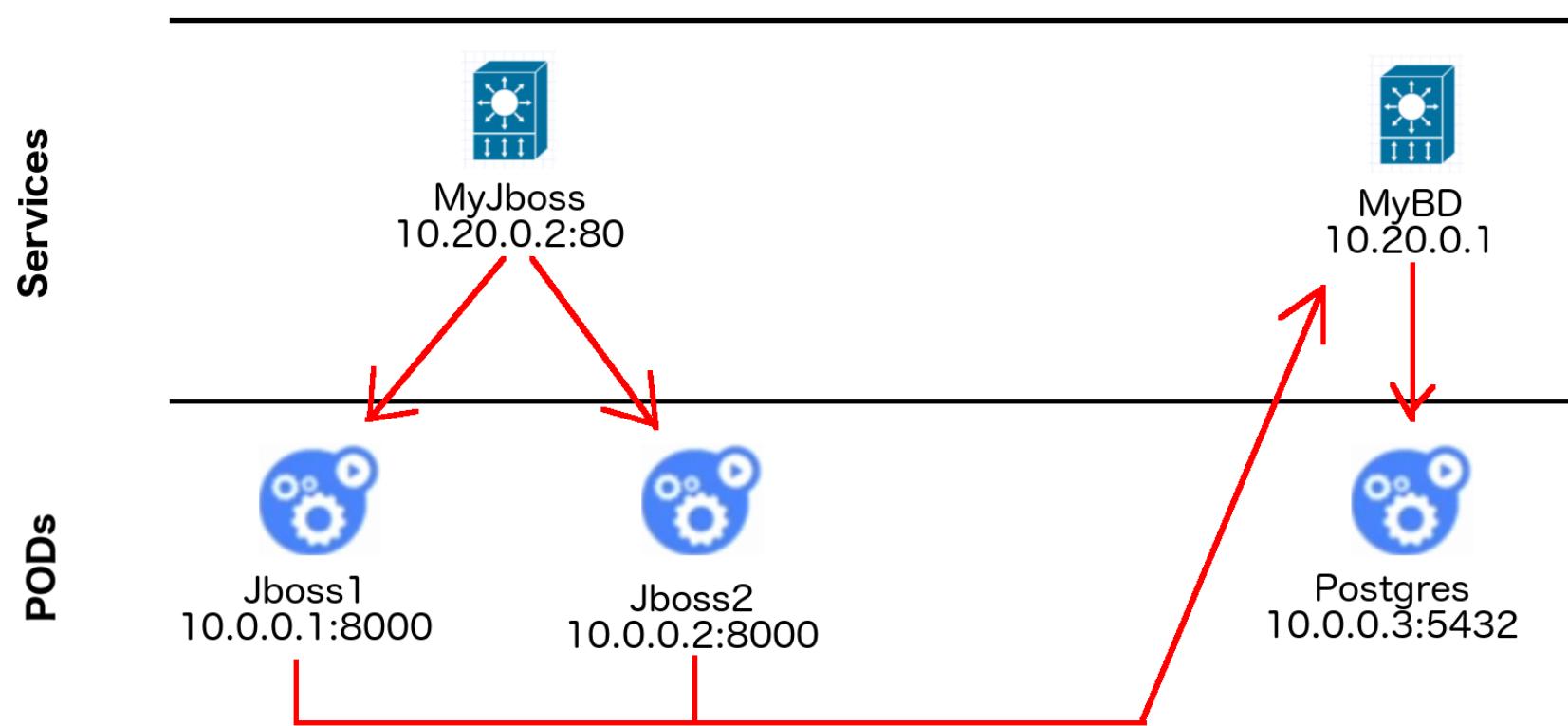


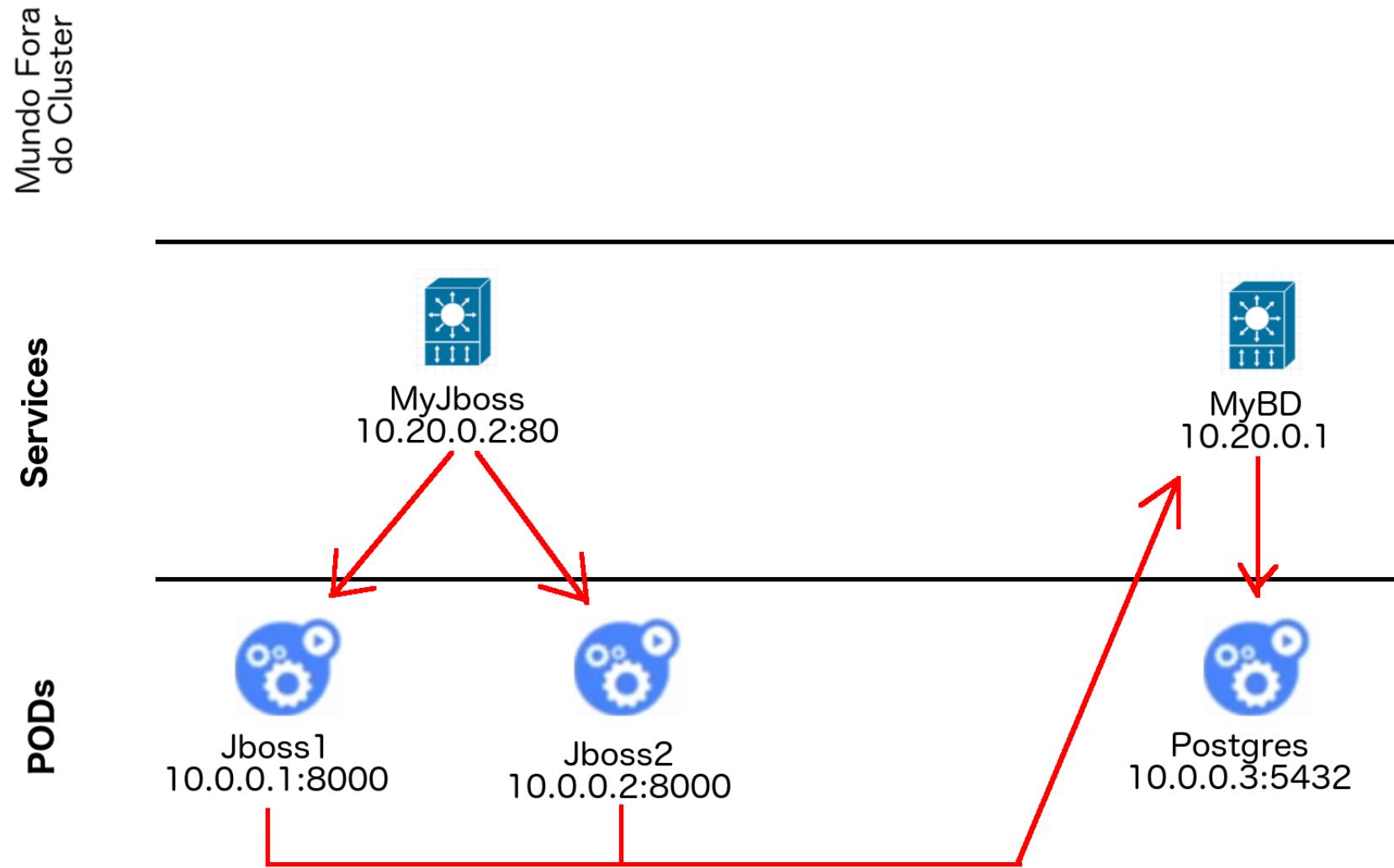


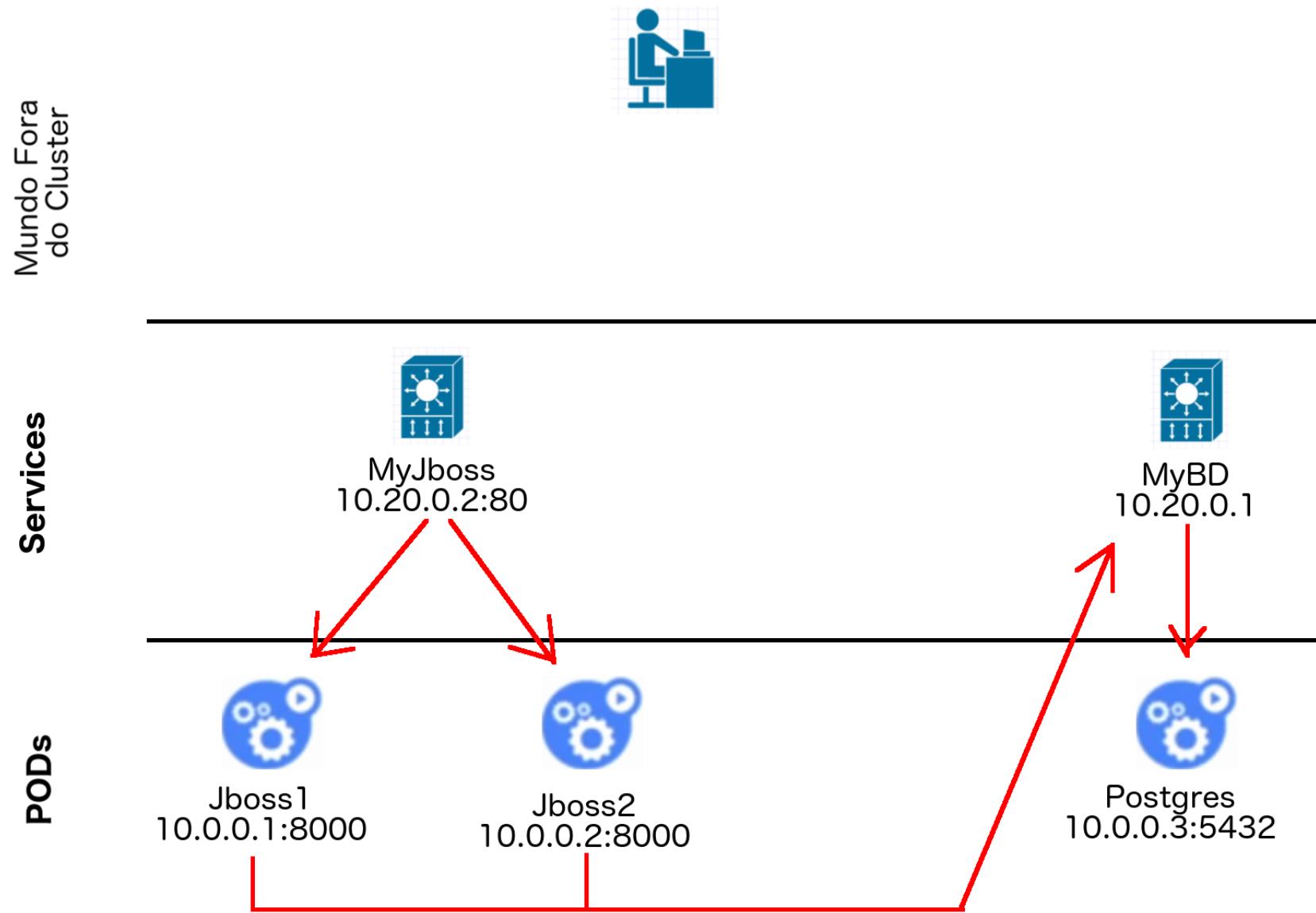


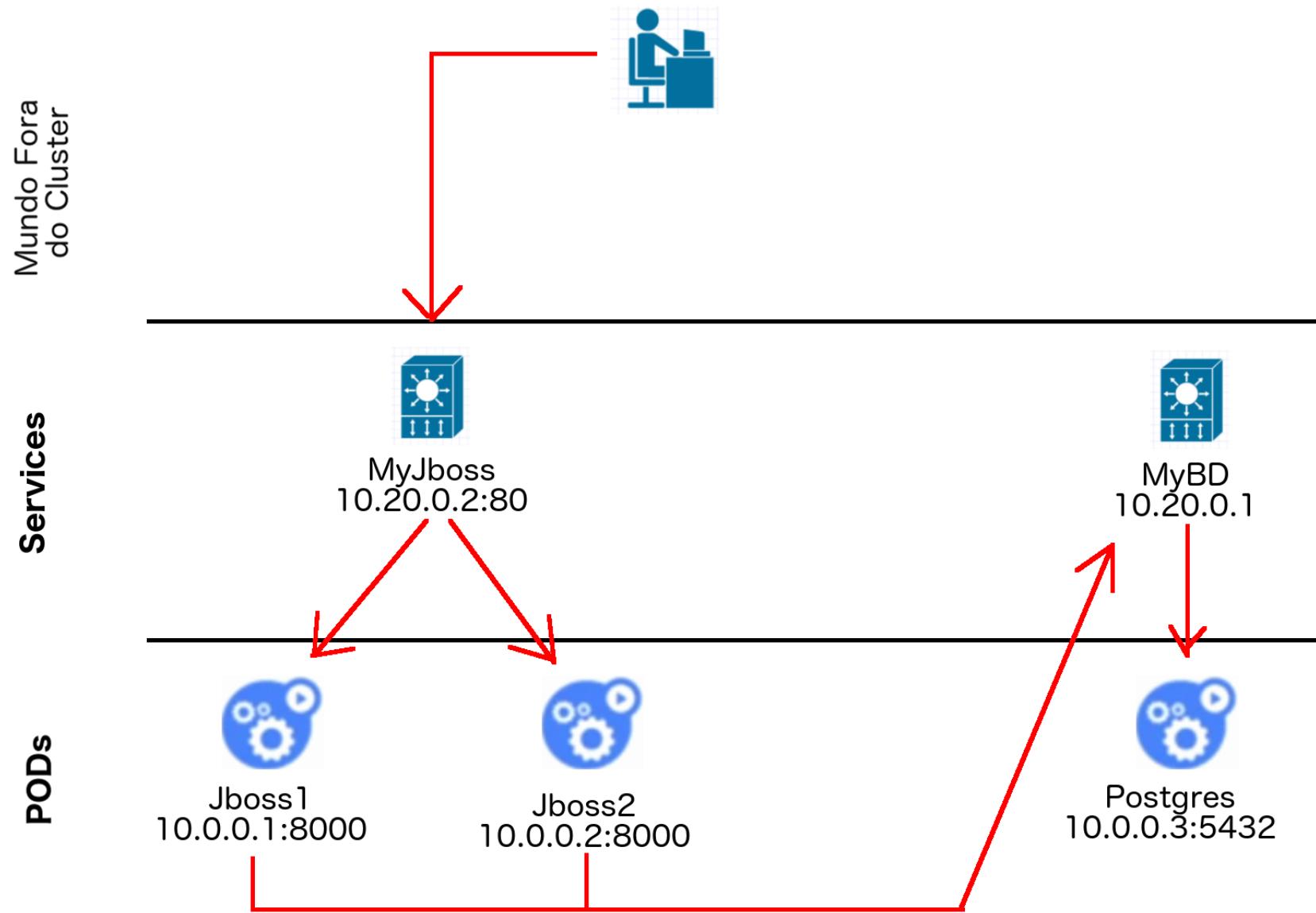


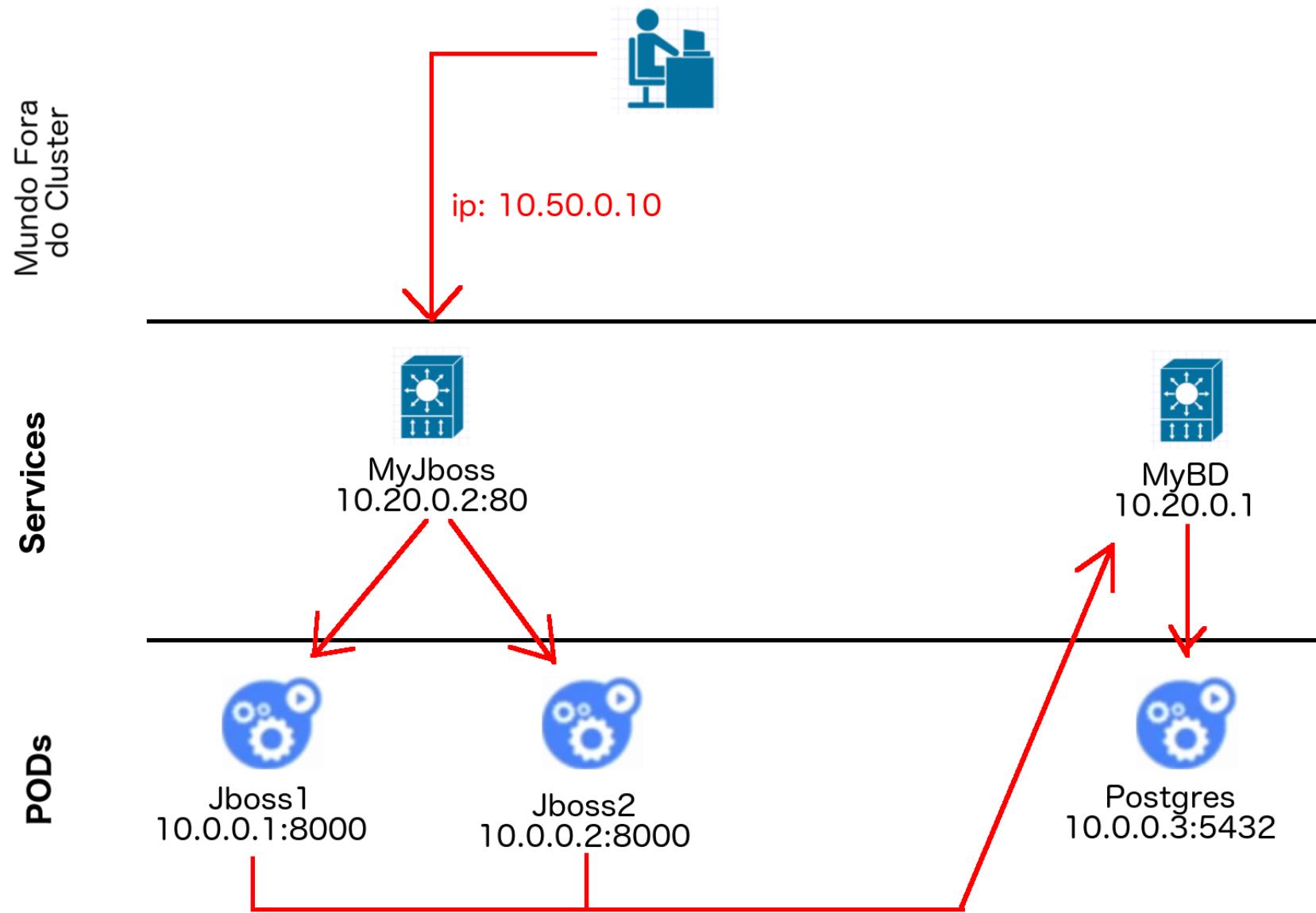


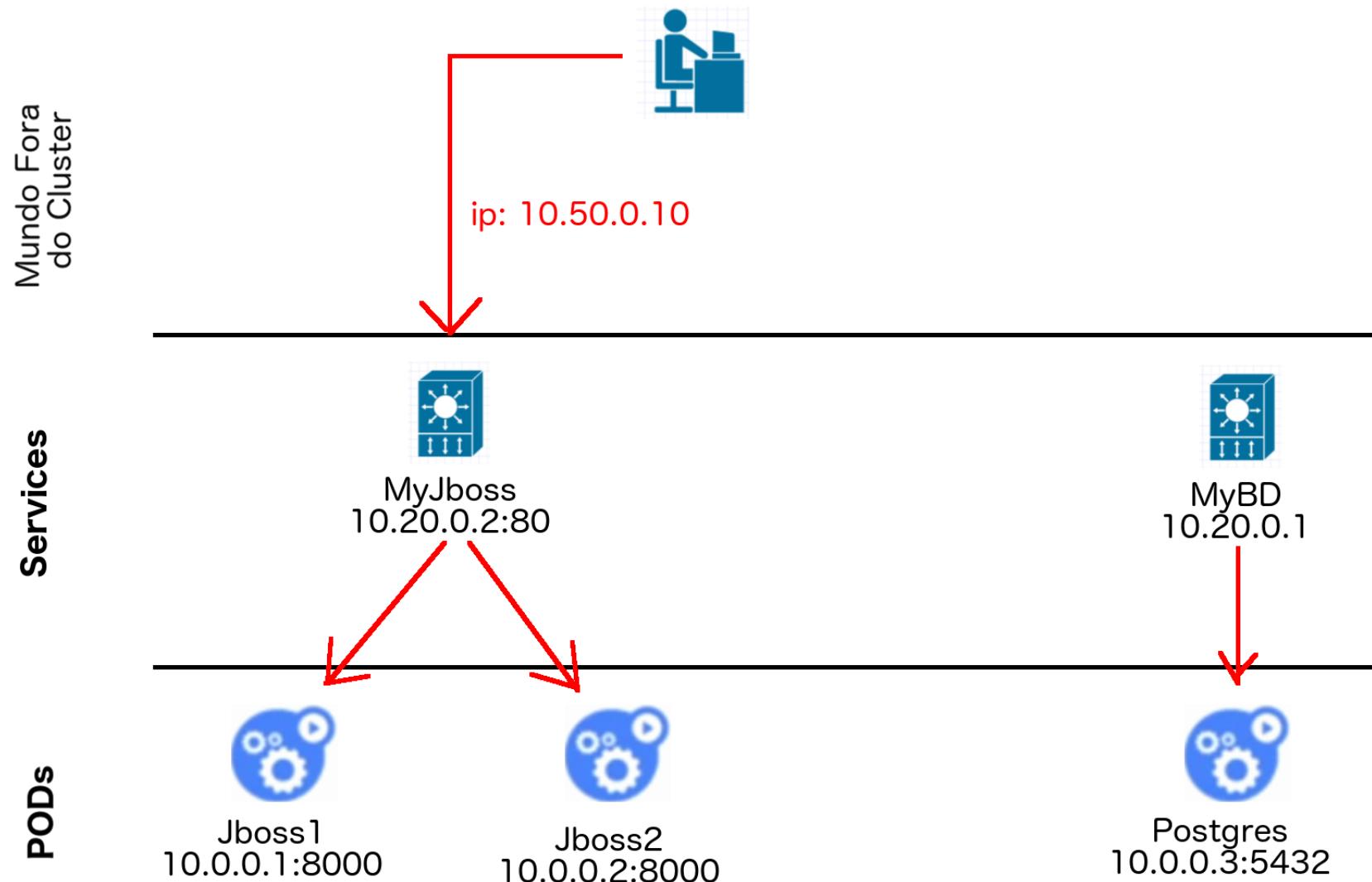


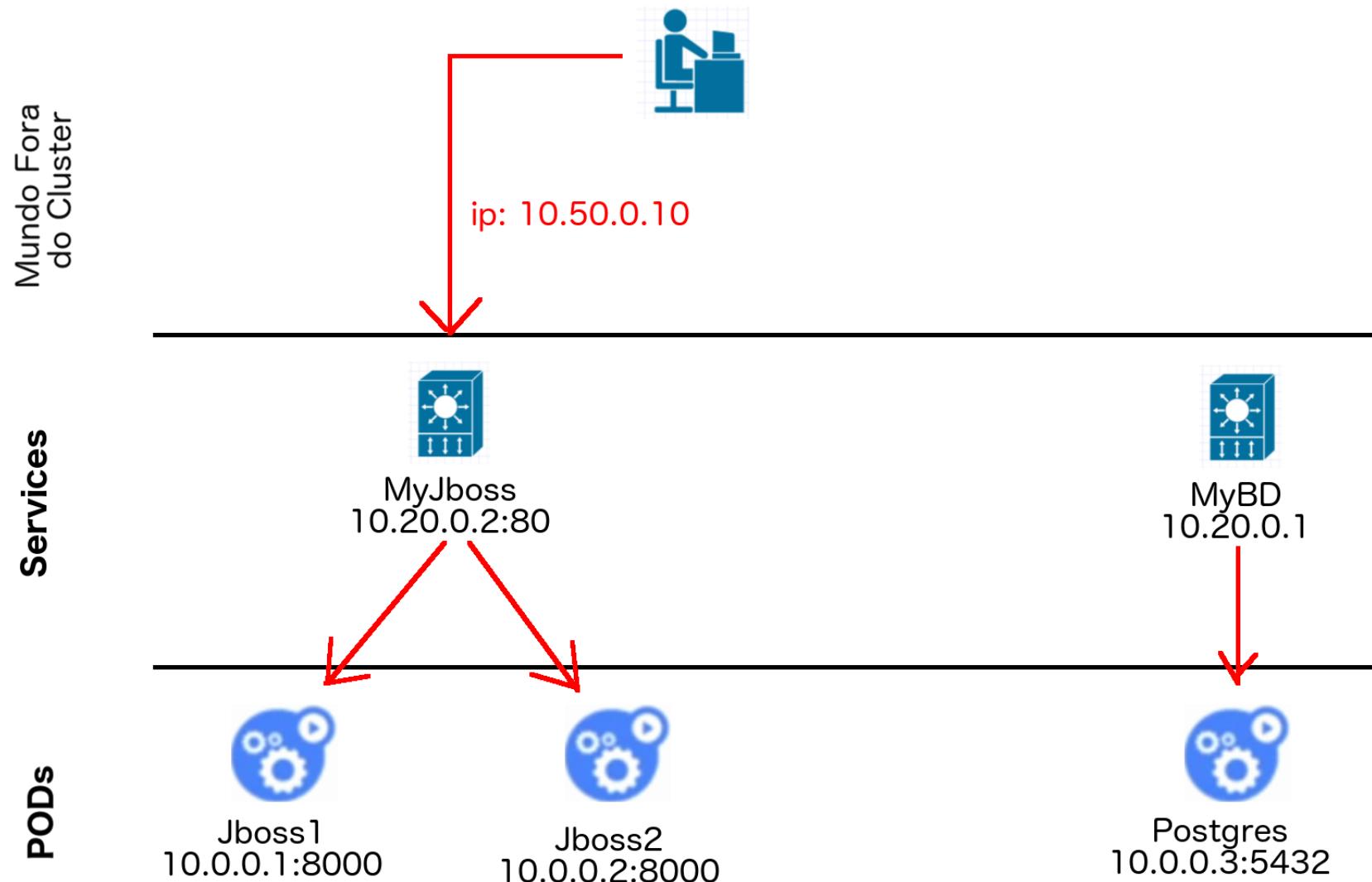


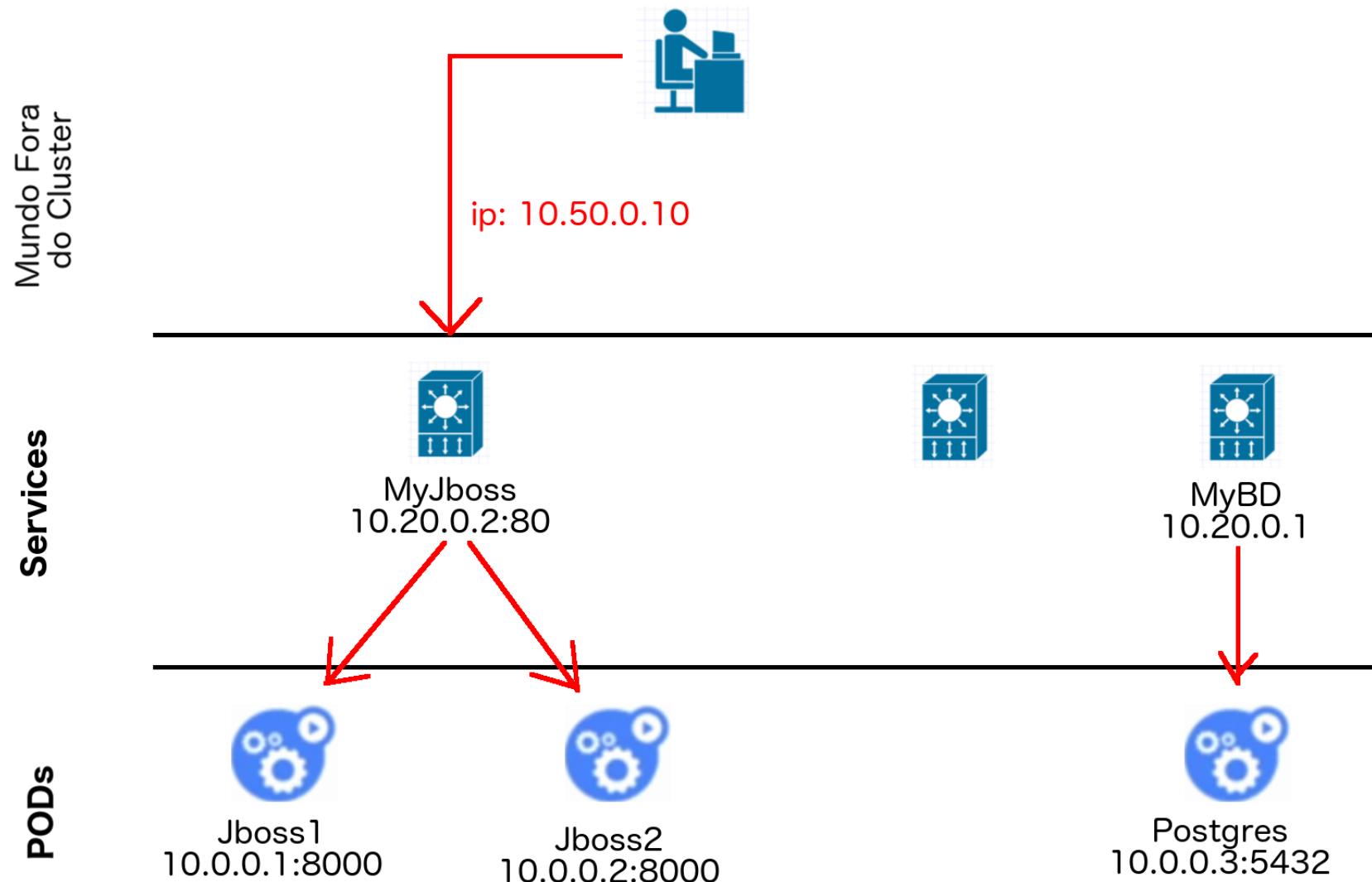


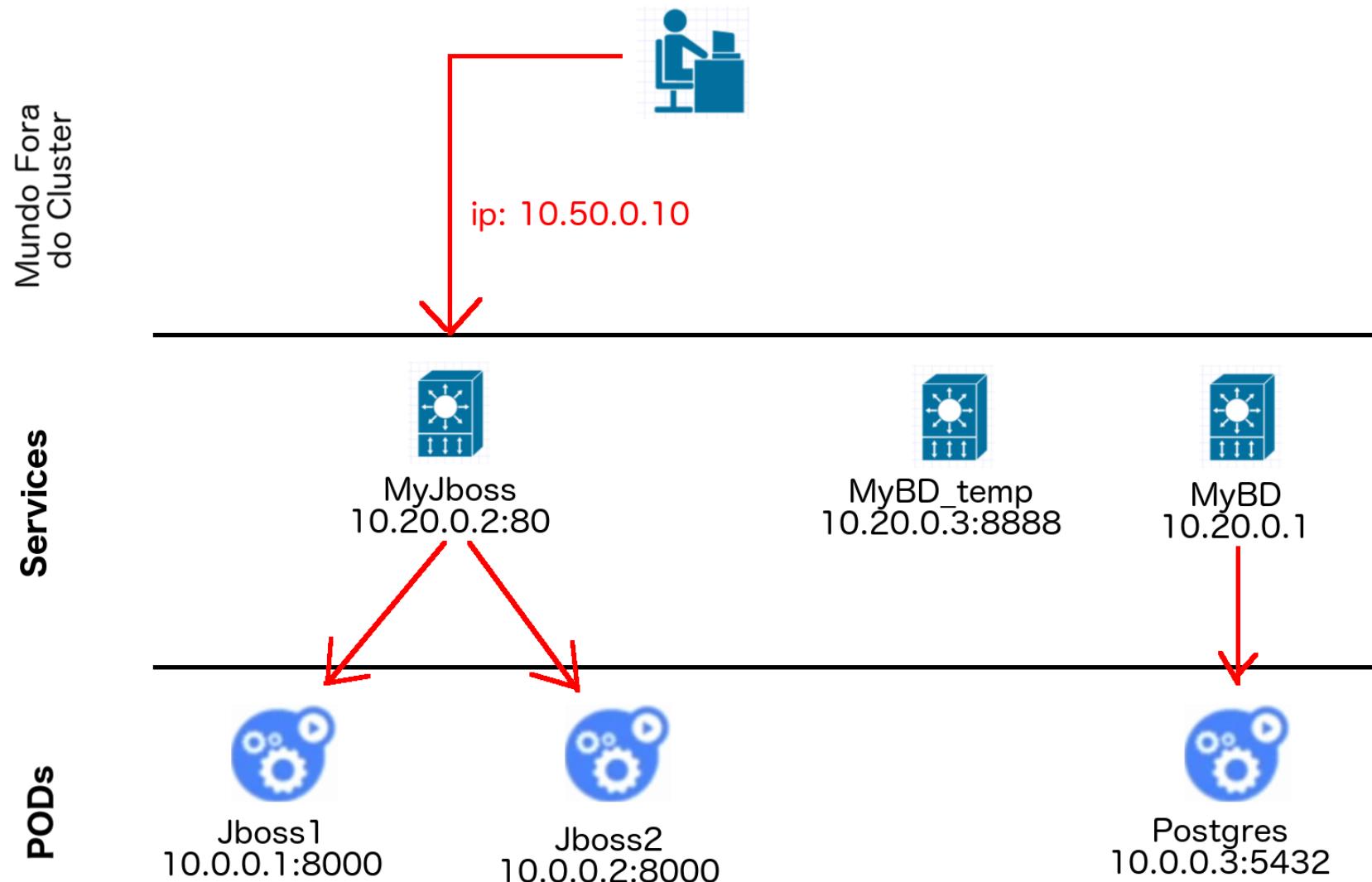


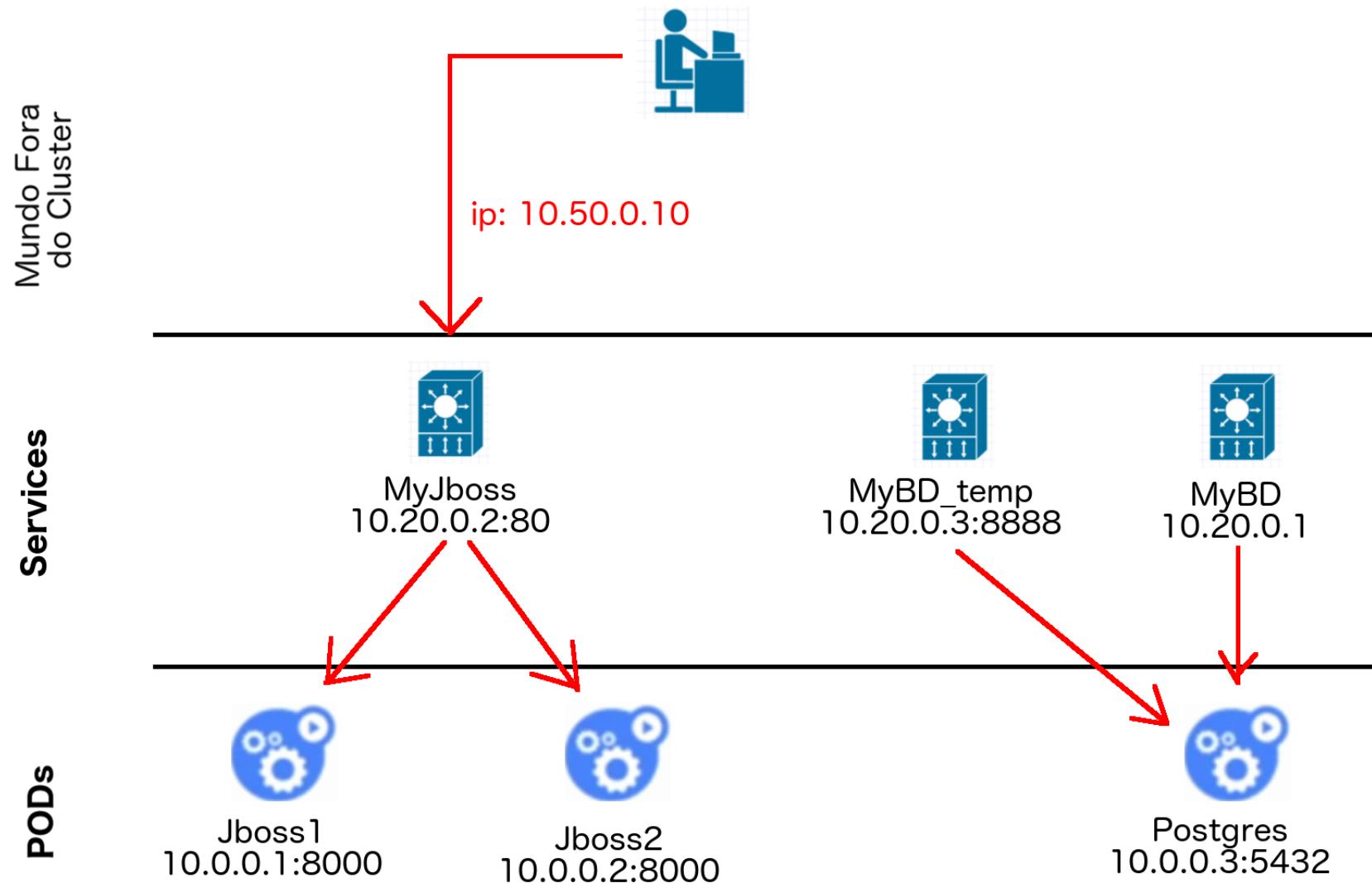


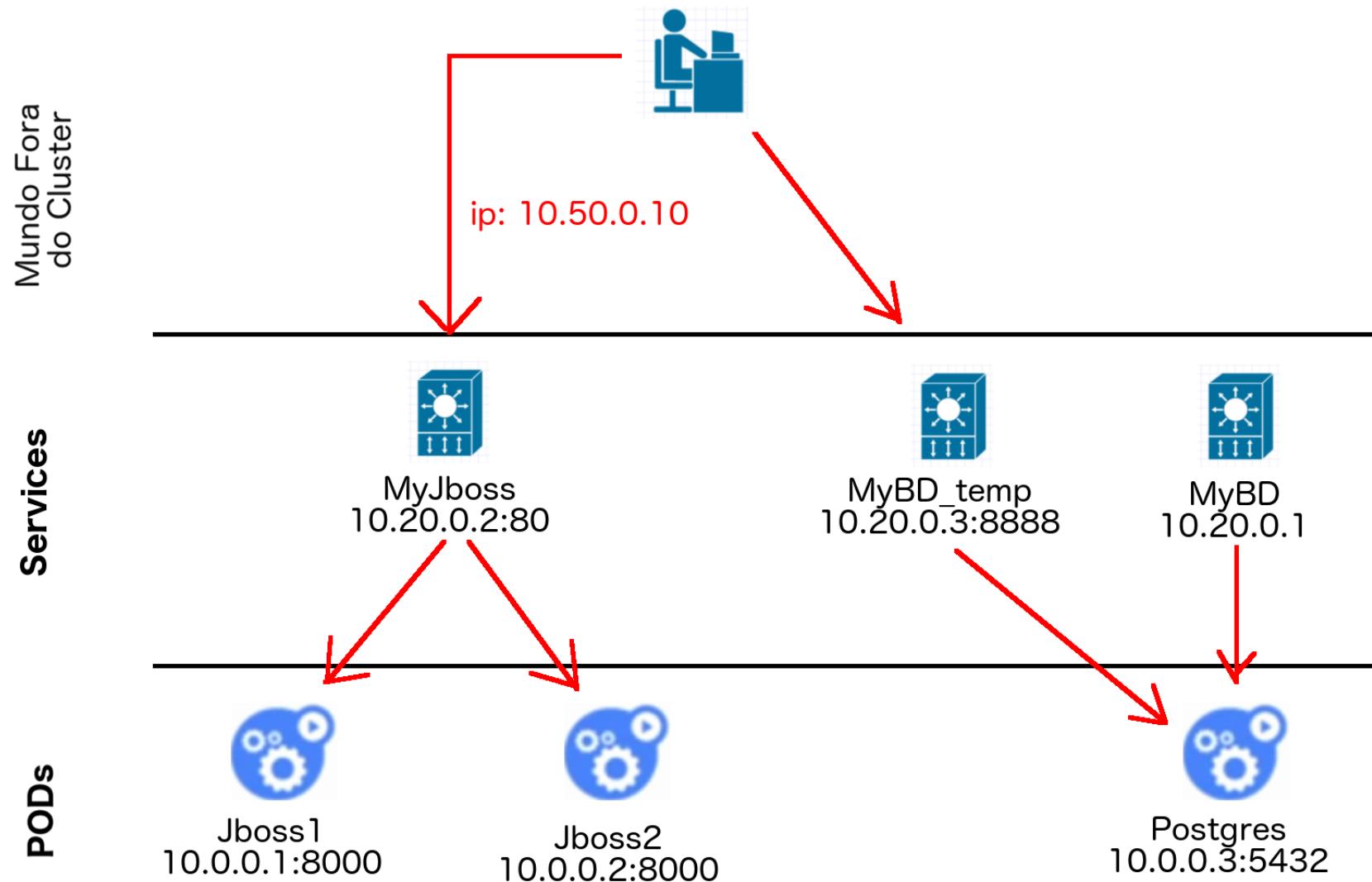


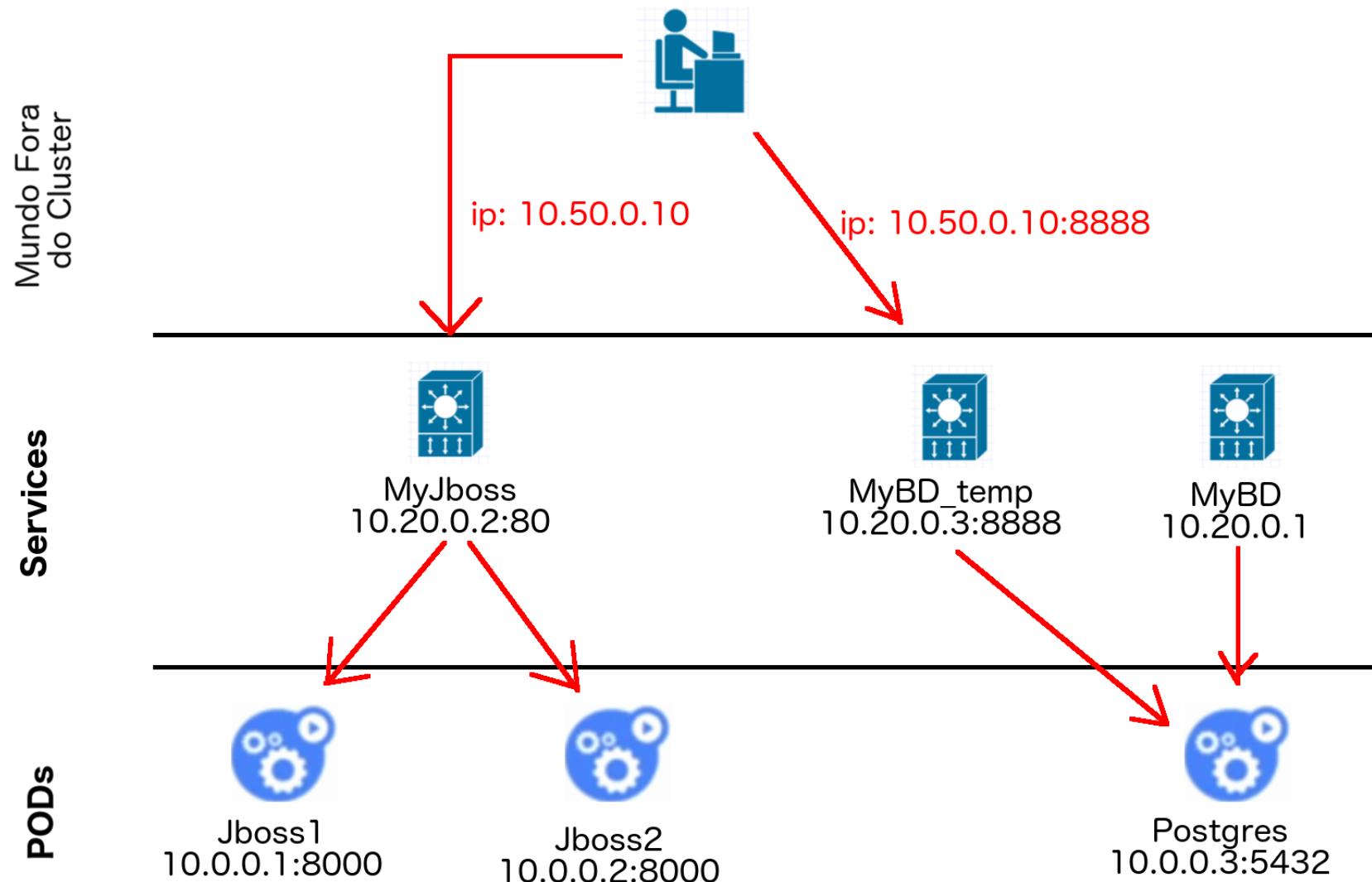












MINISTÉRIO DO PLANEJAMENTO

Demo

OBRIGADO!

fabio.nitto@planejamento.gov.br
marcelo.linhares@planejamento.gov.br