



# NetDevOps

## *Intrarede 2021, Programabilidade das Redes*

*José Miguel Guzmán, Whitestack*

# Desafios da indústria em redes tradicionais

## ¿Quais os desafios que os provedores de serviços e as empresas possuem?



SP  
EMP

**OPEX elevado**  
Horas homem e quantidade de equipamentos



SP  
EMP

**CAPEX elevado**  
Poucas opções e sem economias de escala



SP  
EMP

**Pouca agilidade**  
Nos serviços de rede internos e externos



SP  
EMP

**Aprovisionamento manual e complexo**  
Requerem-se mais especialistas



SP  
EMP

**Equipamentos de função específica**  
Desenhados para o propósito



SP  
EMP

**Redes complexas**  
Milhares de padrões, baixa previsibilidade

# Desafios da indústria em redes tradicionais

## ¿Quais os desafios que os provedores de serviços e as empresas têm?



SP  
EMP

**Dificuldade para escalar no datacenter**  
Devido ao TCO associado



SP  
EMP

**Alta dependência dos fabricantes**  
Impacta no negócio



SP

**Pressão externa de OTTs**  
OTTs e o seu crescimento exponencial do tráfego



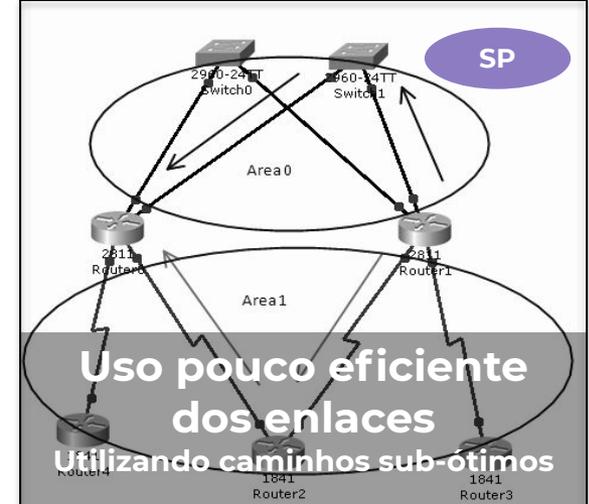
SP

**Pouca inovação nos serviços**  
Cada vez mais difícil concorrer



SP

**Desafios de espaço e energia**  
Na era da explosão do BW



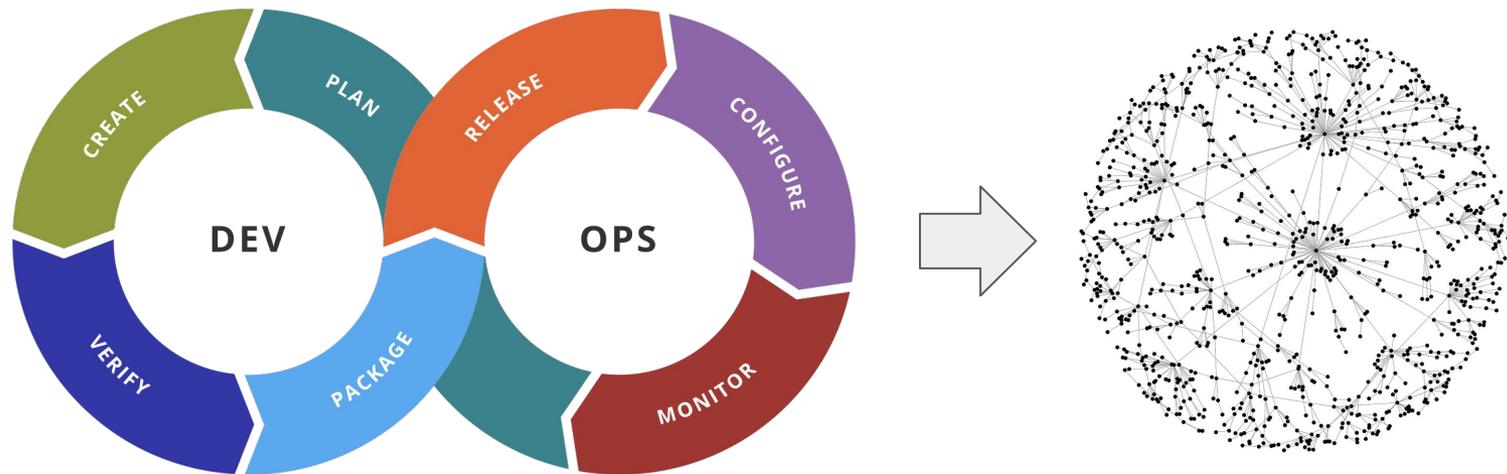
SP

**Uso pouco eficiente dos enlaces**  
Utilizando caminhos sub-ótimos

## Objetivos das novas tendências: DevOps aplicado às redes

# O que queremos resolver com NetDevOps?

DevOps aplicado às redes, ou **NetDevOps**, procura aplicar no mundo das redes as melhores práticas, processos e ferramentas de desenvolvimento de software.



Com o **NetDevOps** queremos **automatizar todos os processos da rede** para **simplificar a operação** da mesma, **alinhado com os objetivos de negócios** da empresa.

# Definindo DevOps e 'NetDevOps'

## Conceito de DevOps

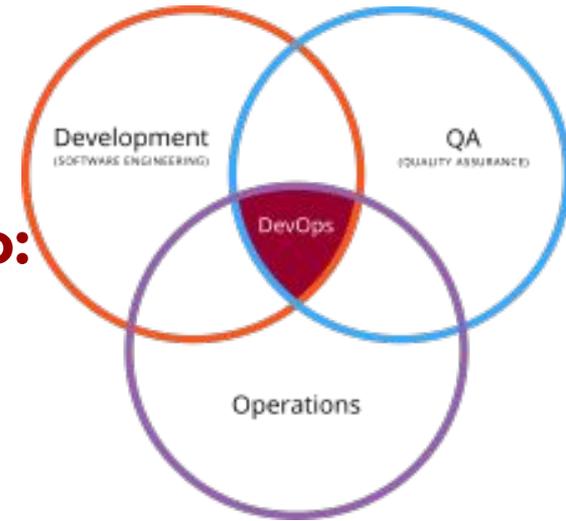


### Técnicas na interseção do

- Desenvolvimento de software
- Quality Assurance
- Operações

para **Acelerar o ciclo de desenvolvimento:**

- Aumenta a frequência de liberação,
- Lançamentos mais estáveis,
- Alinhado com os objetivos do negócio.



### ¿Como?

- Automação
  - Os passos necessários para liberar “Software” (ex: compilação, testes) são automatizados para poder executar o processo de forma estável, múltiplas vezes.
- Monitoração
  - Medir a qualidade da entrega, para quantificar a melhoria introduzida.

## Definindo DevOps e 'NetDevOps'

# Vantagens do DevOps no desenvolvimento de Software

Além das vantagens técnicas, também obtemos:

### Benefícios culturais

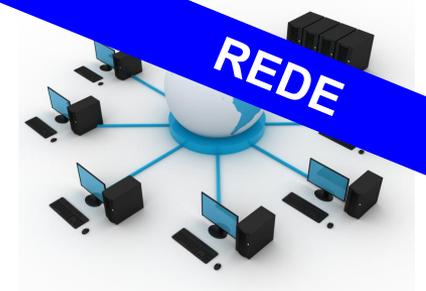
- Times de **desenvolvimento** mais produtivos e felizes
- Maior nível de compromisso
- Maiores oportunidades de crescimento profissional

### Benefícios para o negócio

- Melhor *Time-to-market*
- Comunicação e colaboração otimizadas
- Mais tempo para inovar (no lugar de manter e reparar)

# Definindo DevOps e 'NetDevOps'

## NetDevOps: Levando o DevOps à rede

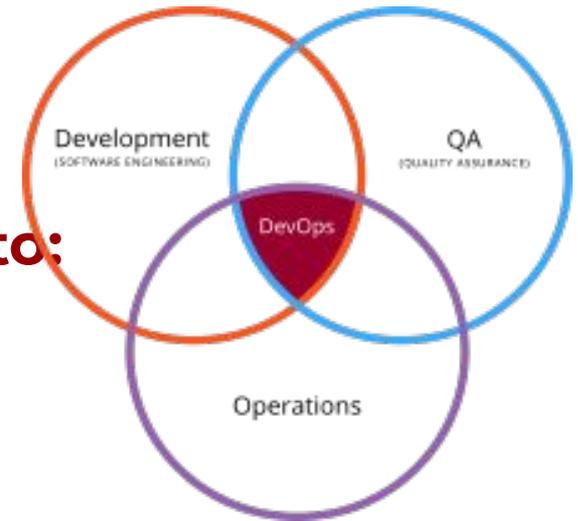


### Técnicas na interseção da

- Criação de Serviços
- Quality Assurance
- Operações

para **Acelerar o ciclo de desenvolvimento:**

- Melhora o *Time-to-Market* de serviços,
- Serviços mais estáveis
- Alinhado com os objetivos do negócio



### ¿Como?

- Automação
  - Os passos necessários para lançar serviços (“Configurar”, “Testar”, “Implementar”) de maneira estável, múltiplas vezes.
- Monitoração
  - Medir a qualidade do serviço, para quantificar a melhoria é introduzida.

## Definindo DevOps e 'NetDevOps'

# Vantagens do NetDevOps na rede

Além das vantagens técnicas, também obtemos:

### Benefícios culturais

- Times de **operações** mais produtivo e felizes
- Maior nível de compromisso
- Maiores oportunidades de crescimento profissional

### Benefícios para o negócio

- Melhor *Time-to-market*
- Comunicação e colaboração otimizadas
- Mais tempo para inovar (no lugar de manter e reparar)

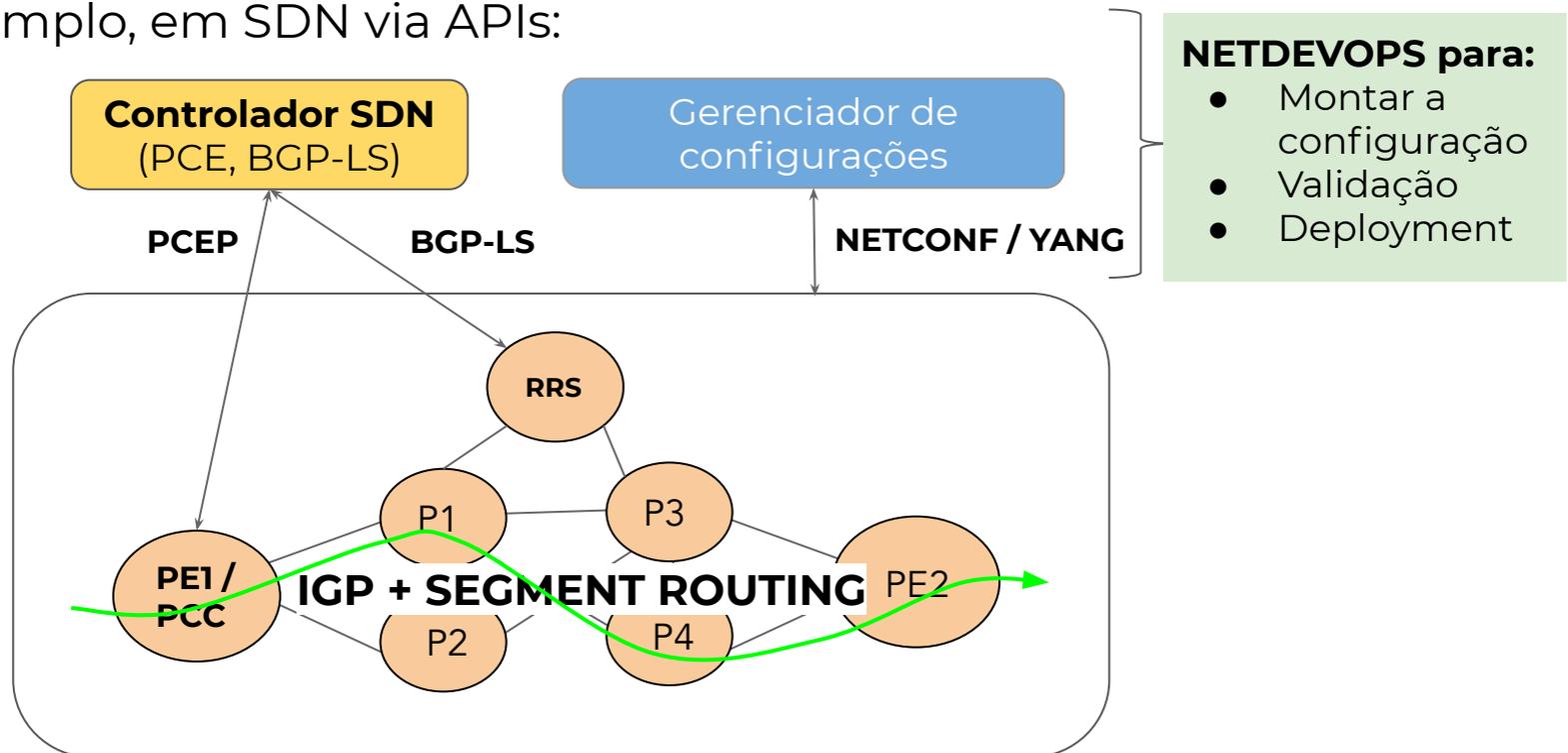
# Definindo DevOps e 'NetDevOps'

## Relação entre NetDevOps e SDN

Há uma relação estreita entre SDN e NetDevops, em particular:

Podemos aplicar técnicas **NetDevOps** em toda interação automatizada com um API de controle ou de gestão de uma **SDN**.

Por exemplo, em SDN via APIs:



## Definindo DevOps e 'NetDevOps'

# Relação entre NetDevOps e SDN

### Net DevOps

É uma pratica as organizações

#### Benefícios para a Organização

- Poupança de custos (TCO)
- Melhor engenharia e operacao
- Eficiência

precisa

### SDN /

### Programabilidade

É uma tecnologia das Redes

#### Benefícios para a Rede

- Poupança de custos
- Melhor engenharia de tráfego
- Eficiência

## Definiendo DevOps y 'NetDevOps'

# Relação entre NetDevOps e SDN

Existem tecnologias e ferramentas para automatizar configurações de múltiplos *vendors* pois conseguem abstrair o plano de gestão com APIs.

Essas ferramentas permitem implementar parte do que o SDN propõe, principalmente em equipamentos tradicionais (SDN via API)



ANSIBLE



NAPALM



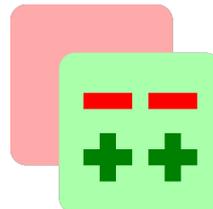
Jenkins



CHEF™



git



## Princípios e práticas relevantes

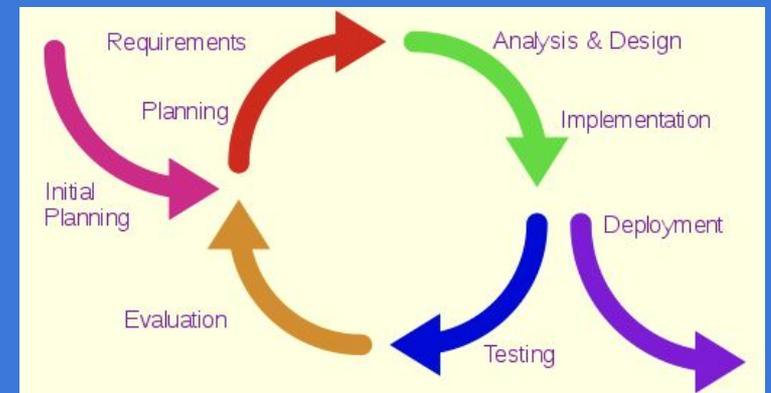
# Princípios DevOps (1)

O DevOps é baseado em **princípios** de pensamento os quais devem ser considerados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Iterações incrementais

Repetir o processo em pequenas porções do problema até atingir os objetivos.



## Princípios e práticas relevantes

# Princípios DevOps (2)

O DevOps é baseado em **princípios** de pensamento os quais devem ser considerados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Continuidade

Executar validações, integrações, implementações e desenvolvimentos tão frequentemente e tão cedo quanto for possível



## Princípios e práticas relevantes

# Princípios DevOps (3)

O DevOps é baseado em **princípios** de pensamento os quais devem ser considerados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Automação

Automação e monitorização são vitais para cada uma das etapas que compõem o processo de construção do software



# Princípios e práticas relevantes

## Princípios DevOps (4)

O DevOps é baseado em **princípios** de pensamento os quais devem ser considerados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Auto-serviço

Cada indivíduo deve ser capaz de ter os recursos necessários durante todo o processo de forma autogerida.



## Princípios e práticas relevantes

# Princípios DevOps (5)

O DevOps é baseado em **princípios** de pensamento os quais devem ser considerados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Colaboração

O trabalho e a criação são levadas ao mesmo tempo. Um objetivo claro, um time diversificado e comunicação contínua são fundamentais.



# Princípios e práticas relevantes

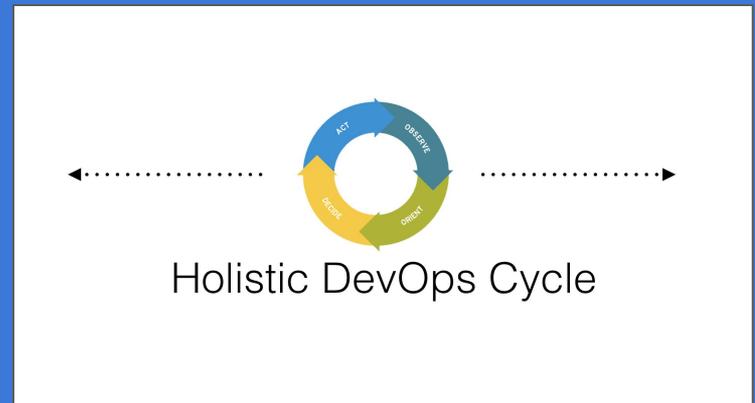
## Princípios DevOps (6)

O DevOps é baseado em **princípios** de pensamento os quais devem ser focados em todo o ciclo de vida de desenvolvimento do software.

Os princípios documentados variam, mas incluem conceitos como:

### Visão Holística

Não perder de vista o objetivo geral, o quadro completo de cada iteração e do projeto.



## Princípios e práticas relevantes

# Práticas de DevOps (1)

Os princípios de DevOps resultam em práticas, que são os tipos de atividades que é necessário executar durante o desenvolvimento.

As práticas variam ligeiramente entre diferentes documentações ou opiniões, as principais são:

### **Práticas relacionadas com a preparação do ambiente**

#### **Configuração autogerida**

A gente deve preparar o ambiente de desenvolvimento.

#### **Aprovisionamento automatizado**

O ambiente deve estar disponível sob demanda, automaticamente.

## Princípios e práticas relevantes

# Práticas de DevOps (2)

Os princípios de DevOps resultam em práticas, que são os tipos de atividades que é necessário executar durante o desenvolvimento.

As práticas variam ligeiramente entre diferentes documentações ou opiniões, as principais são:

### **Práticas relacionadas com a continuidade dos processos**

#### **Construção Contínua (continuous build)**

Construir código o mais frequente e o mais cedo possível

#### **Integração Contínua (continuous integration)**

Integrar o código com maior frequência e o mais cedo possível

#### **Entrega Contínua (continuous delivery)**

Lançar o código finalizado com maior frequência e o mais cedo

## Princípios e práticas relevantes

# Práticas de DevOps (3)

Os princípios de DevOps resultam em práticas, que são os tipos de atividades que é necessário executar durante o desenvolvimento.

As práticas variam ligeiramente entre diferentes documentações ou opiniões, as principais são:

### Práticas pós-lançamento

#### **Gerenciamento de versões automatizada**

Uso de ferramentas que automatizam o lançamento das versões

#### **Validação incremental (incremental testing)**

Validar o código com cada iteração incremental



# Princípios e práticas relevantes

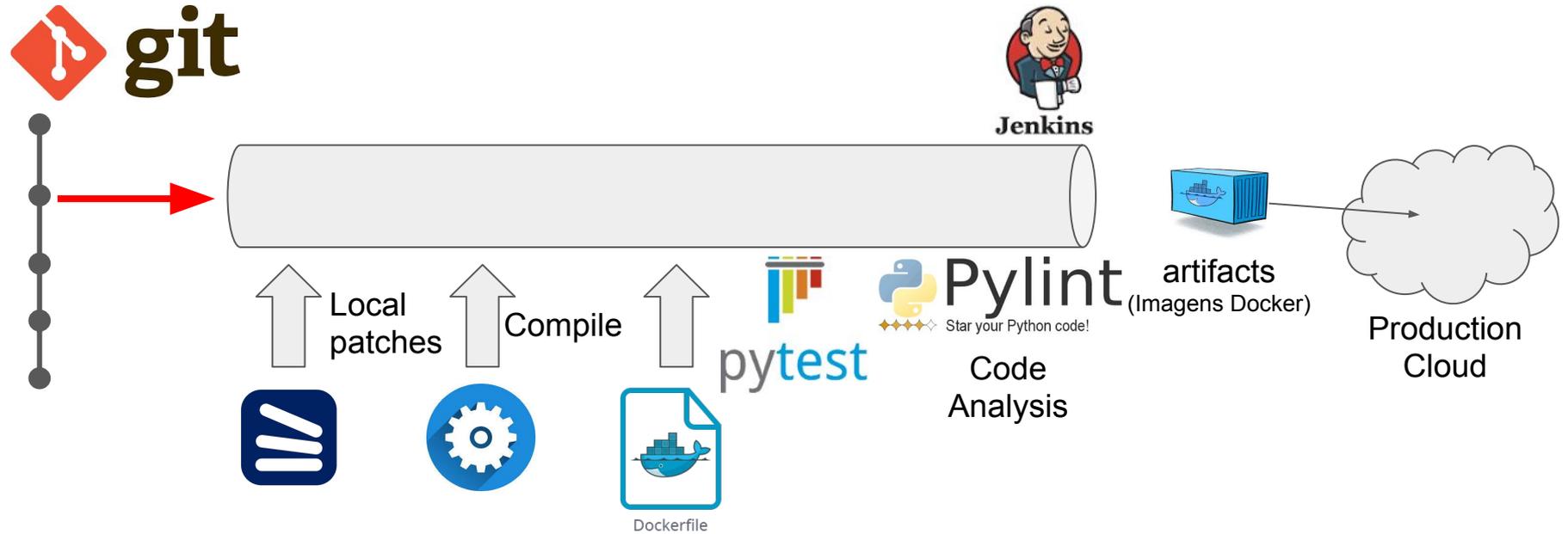
## Exemplo de prática: CI/CD (Software) Continuous integration & Delivery

Permite “construir” Software automaticamente:

- Compilar, Testar, Gerar documentação, Gerar imagens, etc.

Permite “implantar” Software em Produção automaticamente

- Copiar artefatos, instalar, reiniciar, testar.





# Princípios e práticas relevantes

## CI/CD (Red)

## Continuous Integration & Delivery

Permite “construir” serviços automaticamente

- Validar Configurações/Playbooks

Permite “implantar” Serviços automaticamente

- Configurar na rede
- Experimentar o serviço



Verificar



Inserir  
Parâmetros



Aprovar



Jenkins



ANSIBLE



**USA**  
Whitestack, LLC.  
Brickell Bayview Center  
80 SW 8th Street, Suite 2000,  
Miami, FL 33130

**LATAM**  
Whitestack Chile Spa  
Apoquindo 4700 piso 11  
Las Condes, 7560969  
Chile

Whitestack Perú SRL  
Av. Pardo Nro. 434 Int. 1602  
Lima, 15047  
Perú

