

# Value Stream Mapping: uma visão geral

Renato Groffe

Microsoft MVP, Docker Captain,  
APISec U Ambassador, MTAC

[linkedin.com/in/renatogroffe](https://linkedin.com/in/renatogroffe)  
[renatogroffe.medium.com](https://renatogroffe.medium.com)



# Renato Groffe

- Microsoft Most Valuable Professional (MVP)
- Docker Captain
- APISEC U Ambassador
- Multi-Platform Technical Audience Contributor (MTAC)
- Arquiteto de Soluções/Software
- +20 anos de experiência na área de Tecnologia
- Community Leader, Autor Técnico e Palestrante

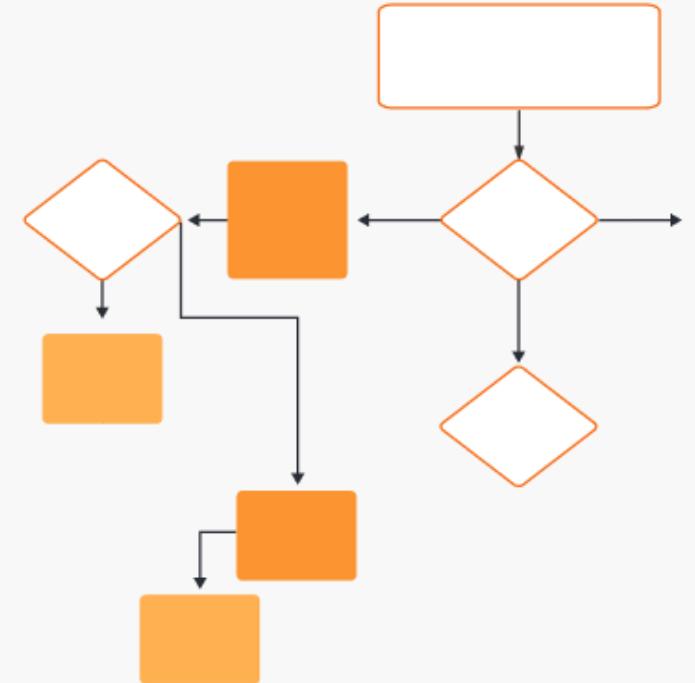


# Iniciativas gratuitas

- Eventos online e gratuitos  
<https://www.meetup.com/dotnet-Sao-Paulo/>

# Agenda

- IT4IT, SAFe...
- O que é Value Stream?
- A importância de DevOps...





How the customer  
explained it



How the Project Leader  
understood it



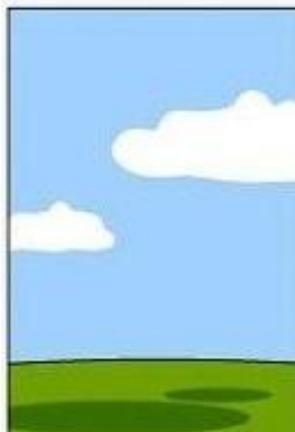
How the System Analyst  
designed it



How the Programmer  
wrote it



How the Business  
Consultant described it



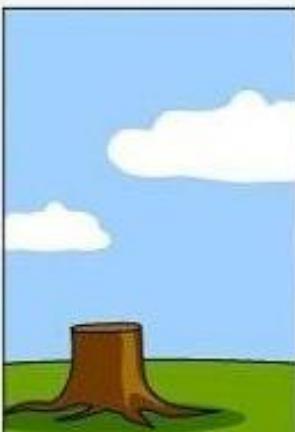
How the project  
was documented



What operations  
installed



How the customer  
was billed



How it was supported



What the customer  
really needed

# O que é IT4IT?

- Framework de arquitetura de referência mantido pela The Open Group
- Foco: integrar e automatizar processos e ferramentas ao longo do **ciclo de vida de serviços de TI**
- Define um "IT Value Chain" com atividades e fluxos de valor claros
- Complementa frameworks como ITIL, COBIT e TOGAF
- Link: <https://www.opengroup.org/it4it>



# O IT Value Chain

Estrutura central com quatro fluxos de valor principais:

1. **Strategy to Portfolio (S2P)** – alinhar estratégia e portfólio de TI
2. **Requirement to Deploy (R2D)** – transformar requisitos em soluções entregáveis
3. **Request to Fulfill (R2F)** – provisionar e entregar serviços de forma eficiente
4. **Detect to Correct (D2C)** – monitorar, detectar e corrigir incidentes

Cada fluxo se conecta por meio de um modelo de informação padronizado



# Benefícios da abordagem IT4IT

- Redução de silos entre equipes e ferramentas
- Aumento de visibilidade ponta a ponta dos serviços de TI
- Automação e integração entre processos (ex.: DevOps + ITSM – Gestão de Serviços de TI)
- Facilita decisões de investimento com base em dados de valor do serviço
- Agilidade e governança equilibradas



# Exemplos práticos no mundo real

- DevOps + ITSM: integração entre ferramentas como Azure DevOps/GitHub Actions e ServiceNow → alinhando pipelines de CI/CD com processos de mudança
- Cloud Management: provisionamento automatizado com Terraform
- Observabilidade: monitoramento com OpenTelemetry em conjunto com soluções como Dynatrace, Datadog, Prometheus, Grafana, Jaeger, Zipkin... conectado ao fluxo "Detect to Correct" para correção proativa
- Gestão de custos: integração de dados financeiros (FinOps) ao fluxo "Strategy to Portfolio" para otimizar investimentos



# Situações onde IT4IT ajuda

- Ambientes multicloud e híbridos → padronizar gestão de serviços
- Organizações em transformação digital → reduzir fricção entre times de negócio, desenvolvimento e operações
- Adoção de automação e IA em TI → conectar monitoramento inteligente ao ciclo de vida do serviço
- Grandes empresas com múltiplos frameworks (ITIL, COBIT, SAFe) → usar IT4IT como "cola" entre eles



# Situações onde IT4IT ajuda

- Ambientes multicloud e híbridos → padronizar gestão de serviços
- Organizações em transformação digital → reduzir fricção entre times de negócio, desenvolvimento e operações
- Adoção de automação e IA em TI → conectar monitoramento inteligente ao ciclo de vida do serviço
- Grandes empresas com múltiplos frameworks (ITIL, COBIT, SAFe) → usar IT4IT como "cola" entre eles



# O que é SAFe?

- Scaled Agile Framework (SAFe®)
- Conjunto de princípios, práticas e papéis para escalar agilidade em grandes organizações
- Baseado em Lean, Agile, DevOps e pensamento sistêmico
- Ajuda a alinhar estratégia, portfólio e execução
- Link: <https://scaledagile.com/>



# Conceitos-chave em SAFe

- Agile Release Train (ART): equipes ágeis trabalhando em sincronia para entregar valor contínuo
- Program Increment (PI): ciclo de planejamento e execução
- Lean Portfolio Management (LPM): governança ágil do portfólio
- Value Streams: estrutura organizacional que conecta estratégia e execução
- DevOps e CI/CD: integrados ao modelo



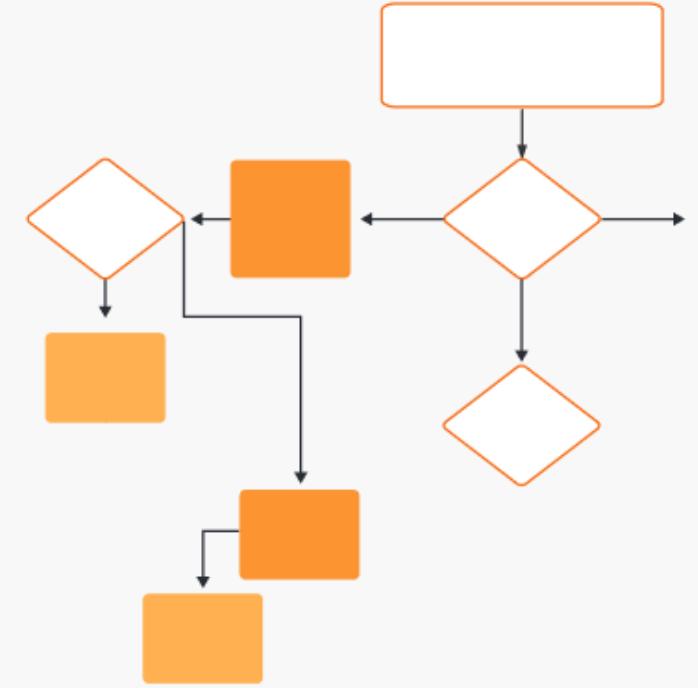
# Benefícios da abordagem SAFe

- Escala práticas ágeis para centenas ou milhares de pessoas
- Maior alinhamento entre TI e negócio
- Transparência e visibilidade do trabalho em todos os níveis
- Melhor **time-to-market** com previsibilidade
- **Governança adaptativa:** decisões de investimento mais rápidas e ajustadas ao valor entregue



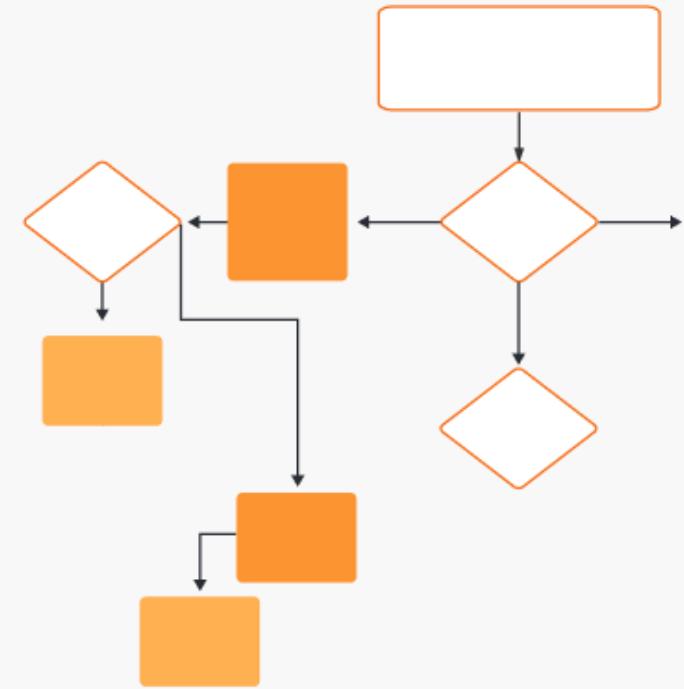
# O que é Value Stream/Fluxo de Valor?

- Conceito originado no **Lean** e ampliado em frameworks como **SAFe** e **IT4IT**
- Representa o **fluxo de atividades necessárias** para entregar valor a um cliente
- Foco no **valor entregue ao usuário final**, não apenas em tarefas internas
- Permite enxergar ponta a ponta (do pedido do cliente até a entrega do valor)
- Link: [https://en.wikipedia.org/wiki/Value-stream\\_mapping](https://en.wikipedia.org/wiki/Value-stream_mapping)

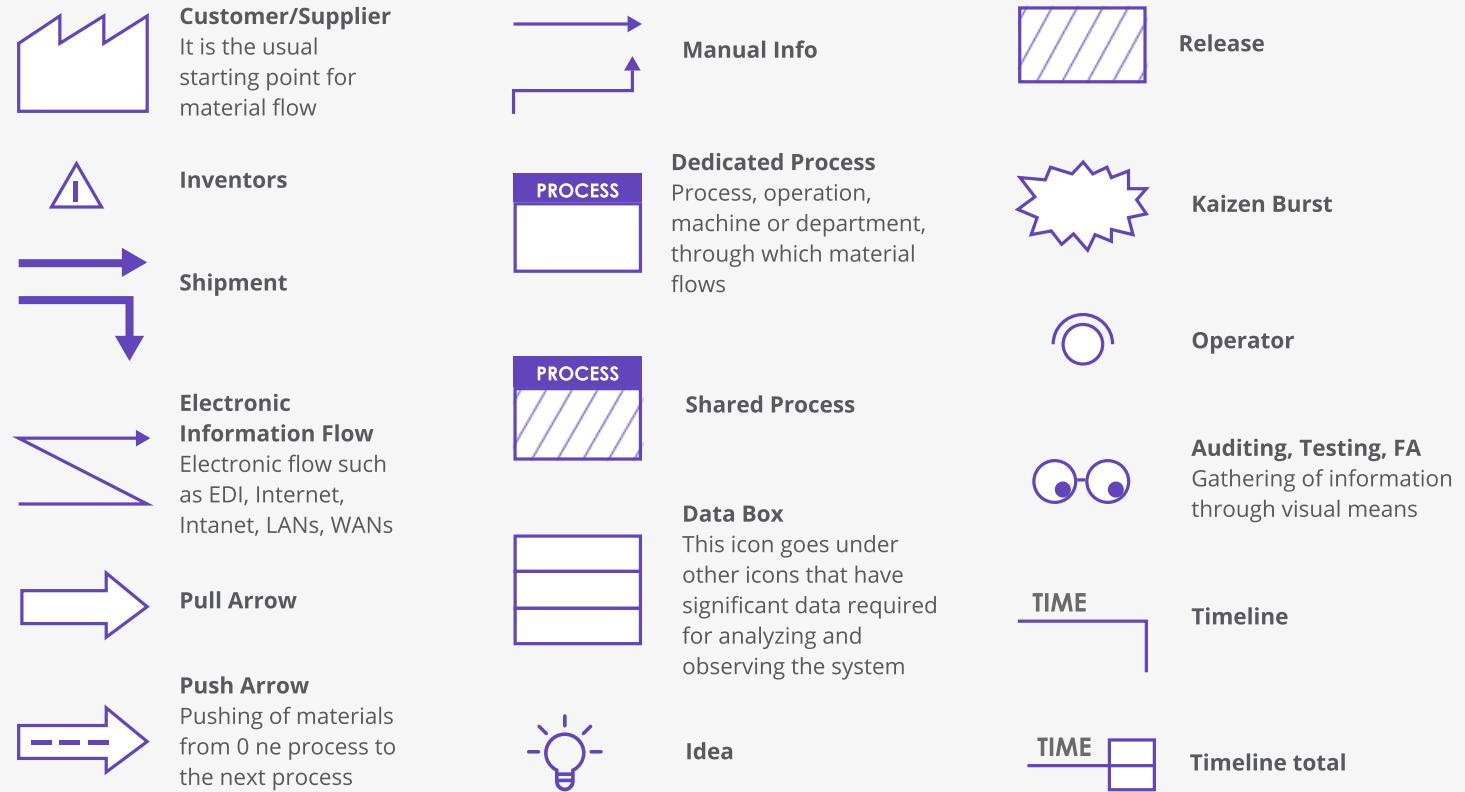


# Estrutura de um Value Stream

- Eventos de gatilho (ex.: cliente solicita um serviço)
- Atividades encadeadas que transformam insumos em valor
- Pontos de espera e desperdício (identificáveis pelo mapeamento)
- Resultado final: valor entregue ao cliente ou ao negócio
- Pode ser operacional (manutenção de serviços) ou de desenvolvimento (criação de novos produtos)

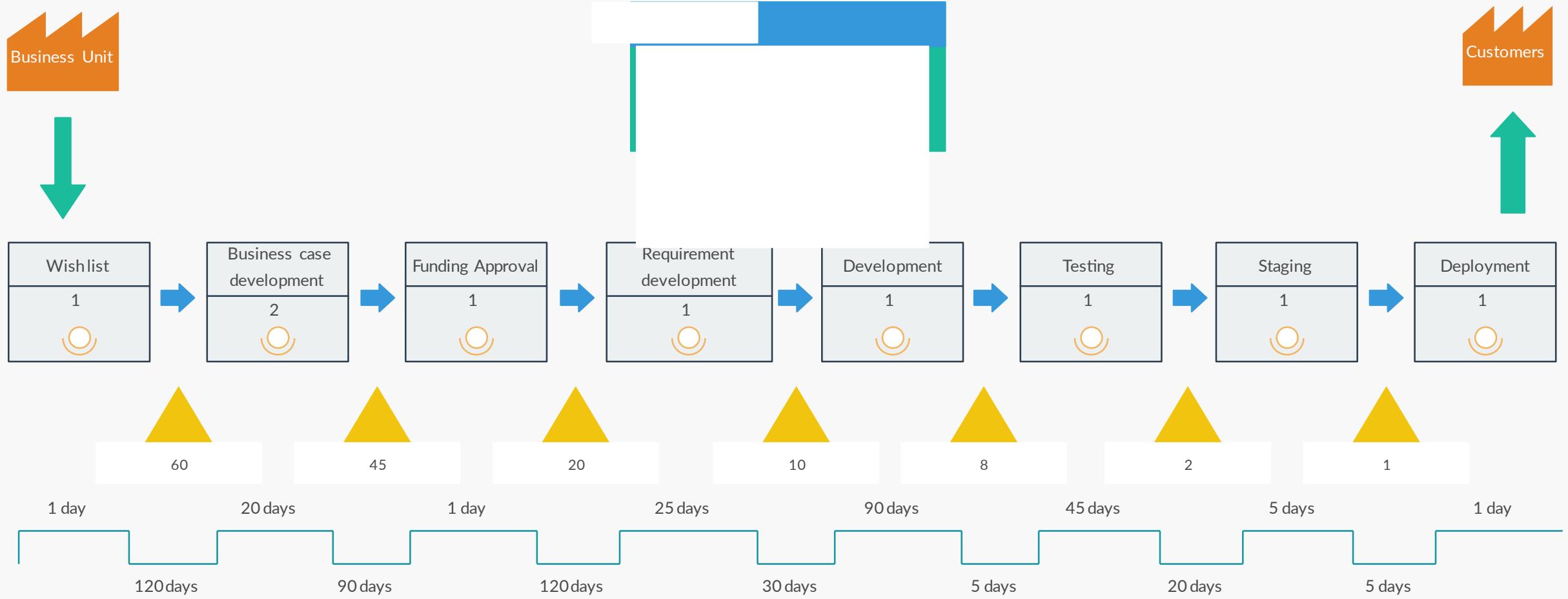


# Alguns elementos de representação



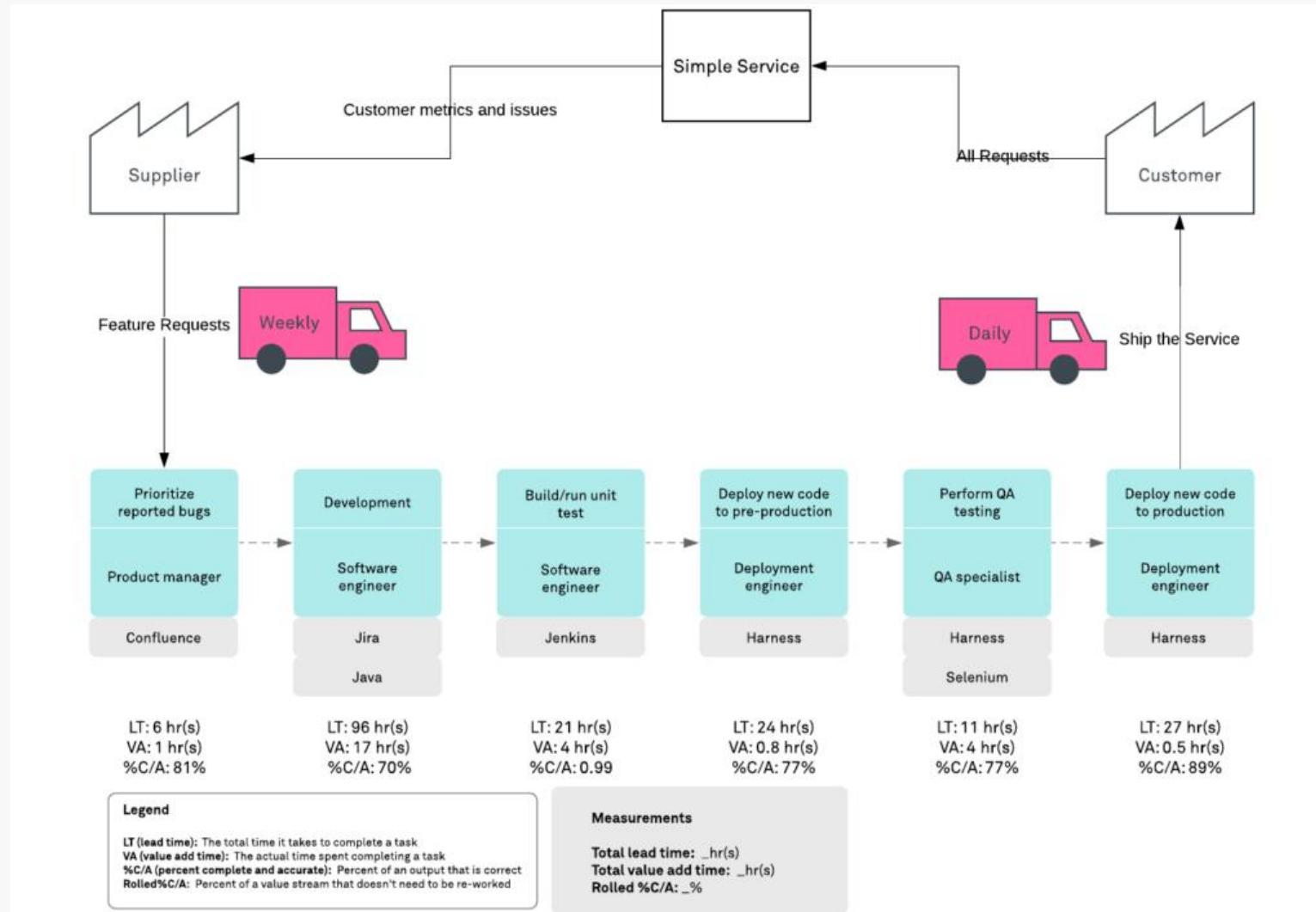
Fonte: <https://big-agile.com/blog/value-stream-mapping>

# Um exemplo...



Fonte: <https://big-agile.com/blog/value-stream-mapping>

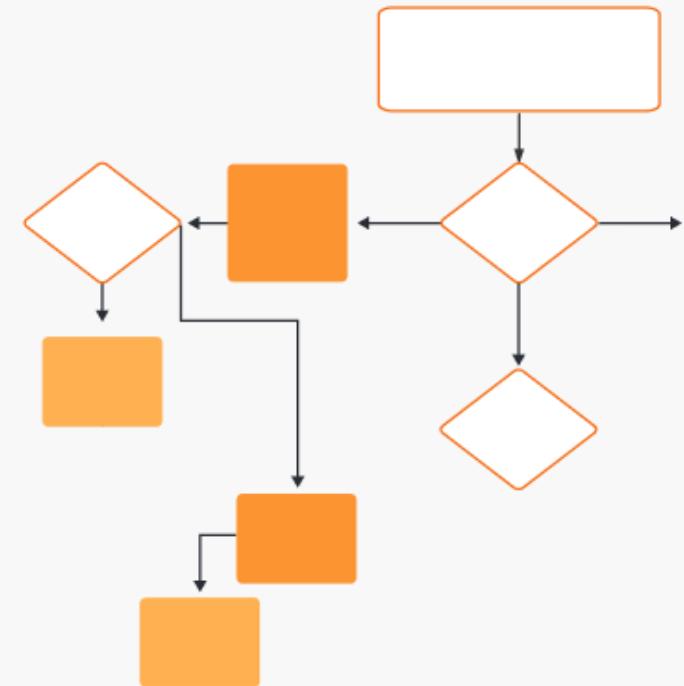
# Um segundo exemplo...



Fonte: <https://www.harness.io/blog/value-stream-mapping-guide>

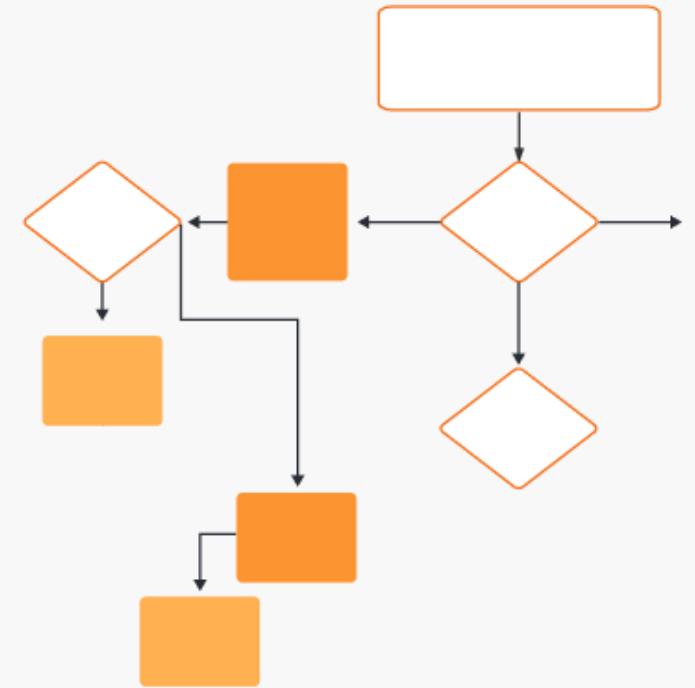
# Benefícios da abordagem

- Clareza sobre como o valor flui na organização
- Identificação de **gargalos, retrabalhos e desperdícios**
- Maior alinhamento entre TI e negócio
- Facilita a priorização de investimentos com base no impacto ao cliente
- Suporte a práticas de agilidade empresarial e DevOps



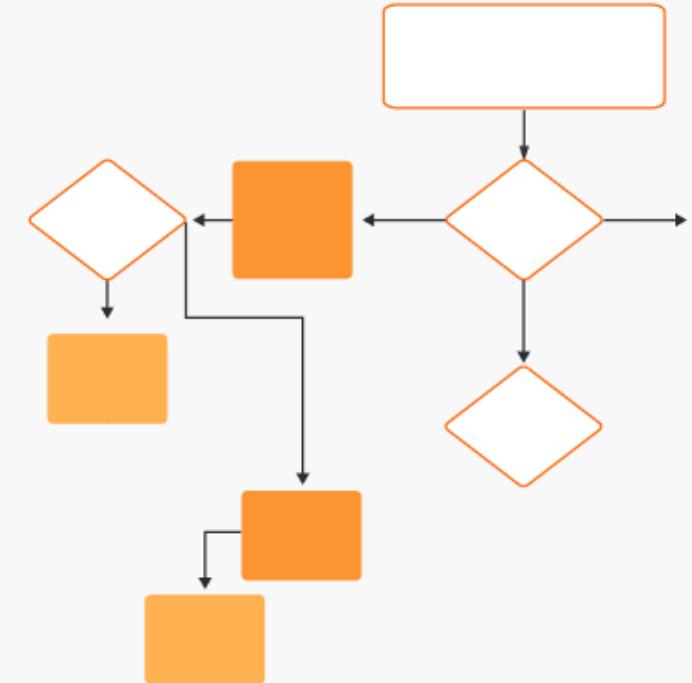
# Exemplos práticos no mundo real

- **E-commerce:** Value Stream de “Pedido a Entrega” → do clique no site até a entrega física do produto
- **Banco digital:** Value Stream de “Abertura de Conta” → cadastro, validação de documentos, aprovação automática, ativação da conta
- **TI corporativa:** Value Stream de “Ideia a Implementação” → do registro de um requisito no backlog até o deployment em produção
- **Saúde:** Value Stream de “Consulta Médica” → agendamento, atendimento, exames, resultado entregue ao paciente



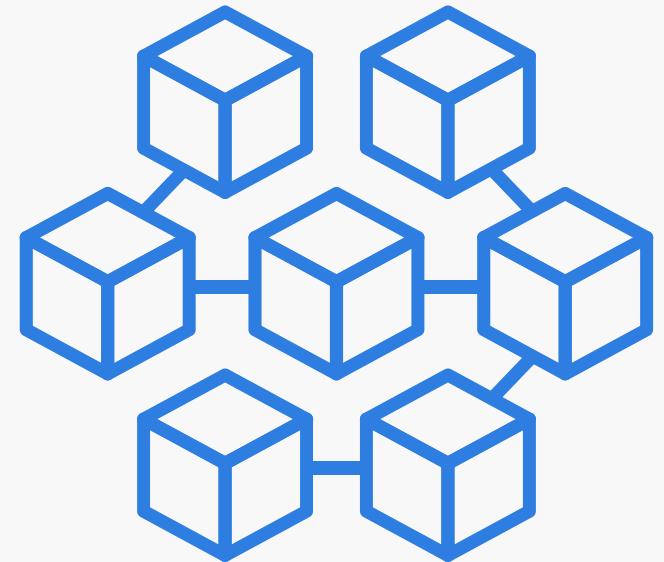
# Situações onde Value Stream pode auxiliar

- Transformação Digital → enxergar o ciclo fim a fim e eliminar burocracia
- Escalando Agilidade (SAFe) → organizar times em torno de fluxos de valor
- DevOps & ITSM → alinhar pipelines de CI/CD com fluxos de valor de entrega de serviço
- FinOps / Custos em TI → identificar onde o valor real é entregue vs. onde há desperdício
- Governança em grandes empresas → conectar estratégia com execução por meio de fluxos de valor

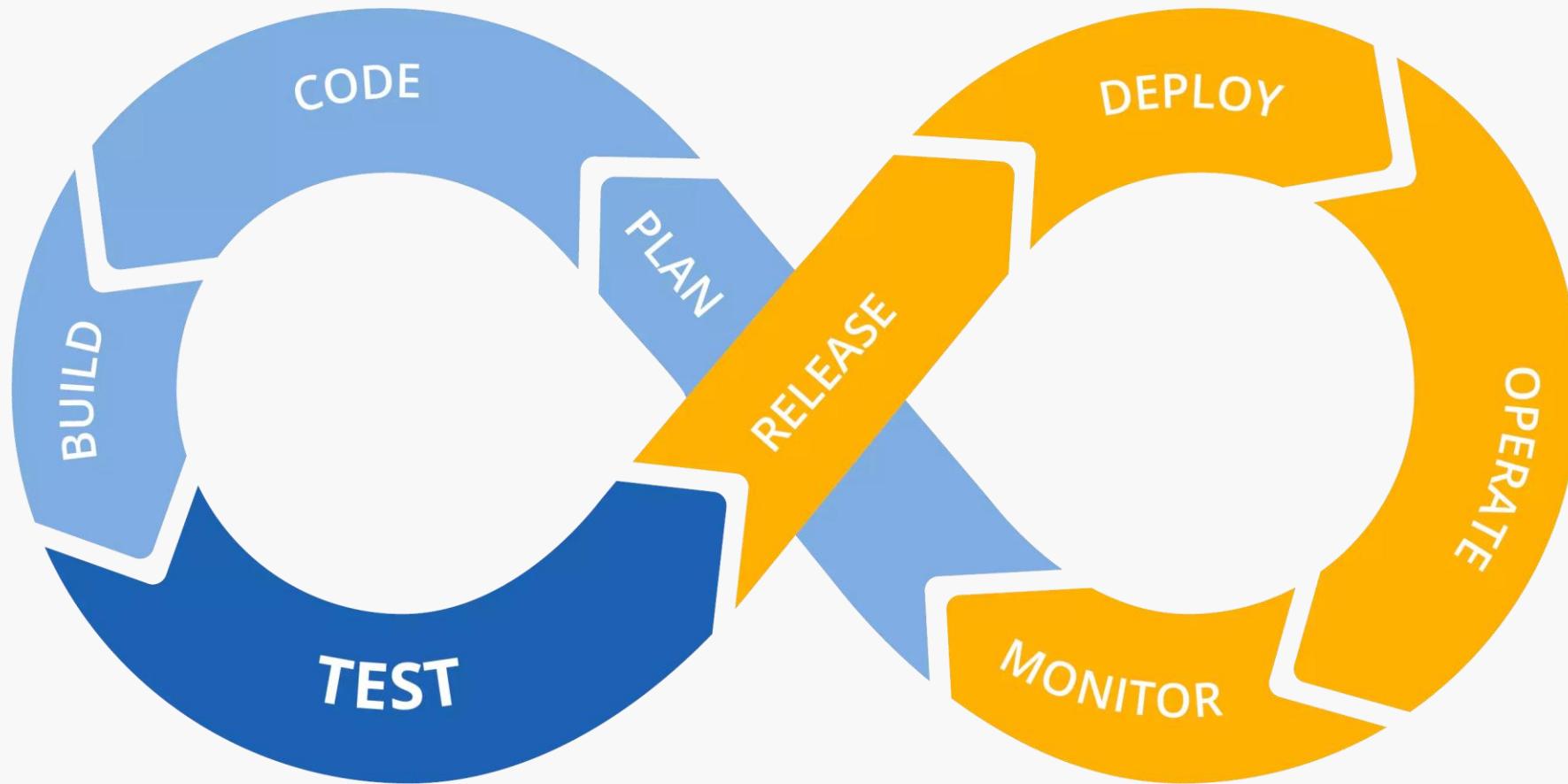


# Problemas comuns em projetos de TI

- Mudanças tecnológicas bruscas, reflexo da falta de planejamento inicial
- Imprevisibilidade
- Apreensão com ambientes de Produção
- Abismo entre requisitos x implementações
- Testes deficientes
- Monitoramento/Observabilidade inexistentes



# DevOps...



# O que realmente é DevOps?

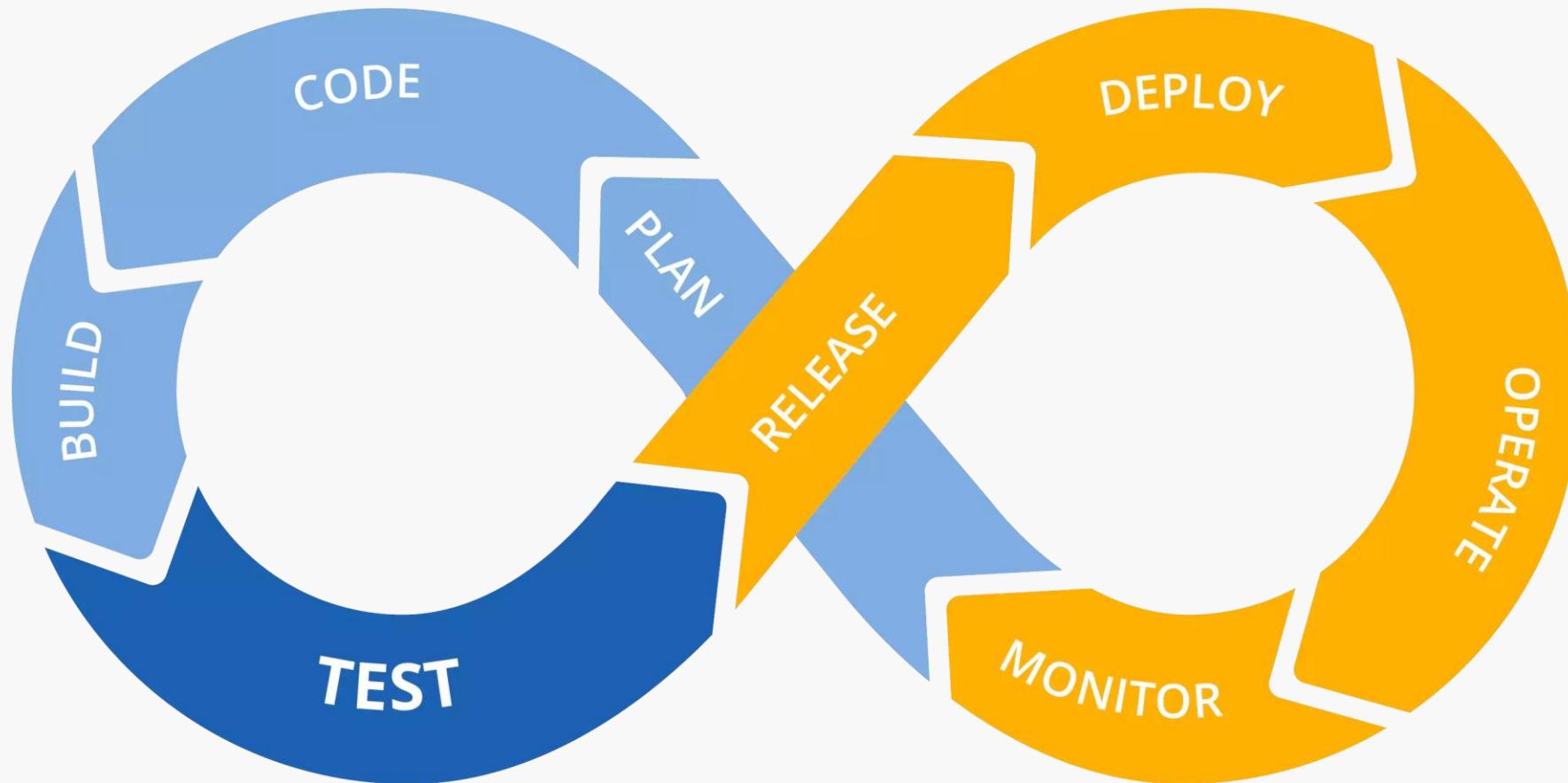
Segundo Donovan Brown (quando atuava na Microsoft):

*DevOps é a união de pessoas, processos e produtos para possibilitar a entrega contínua de valor aos nossos usuários finais.*



Link: <https://devblogs.microsoft.com/devops/what-is-devops-donovan/>

# DevOps + Shift left e Shift right: podem ser uma solução...



## Shift left

Assegurar que requisitos/design são cumpridos

## Shift right

Assegurar performance, resiliência e confiabilidade

# Shift left

- Atender às necessidades do negócio, com uma melhor validação dos requisitos
- Testes validando questões como qualidade, segurança, configurações esperadas e performance no **início** do processo de Desenvolvimento
- Antecipar necessidades e eventuais problemas que ocorreriam em Produção



# Shift right

- Atender às necessidades do negócio, com uma melhor validação dos requisitos
- Testes validando questões como qualidade, segurança, configurações esperadas e performance **em Produção**
- Foco em situações reais



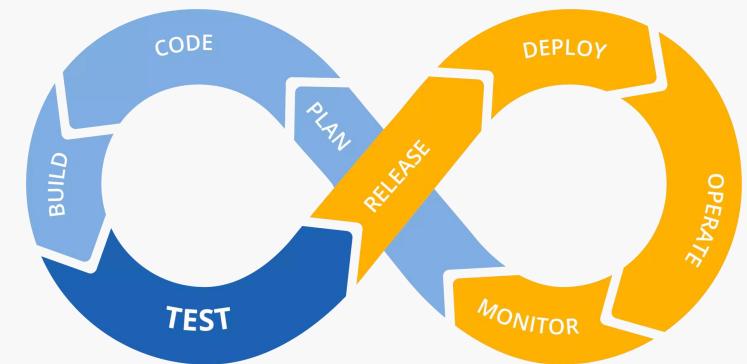
# Shift right: algumas estratégias

- Chaos Engineering
- Synthetic monitoring
- Canary releases, Blue-green deployments



# Considerações importantes

- Faça provas de conceito sempre
- Elabore diagramas e representações da sua arquitetura, em conformidade com requisitos de negócio
- Utilize containers e Infra como Código (IaC)
- Automatize seus testes desde o início

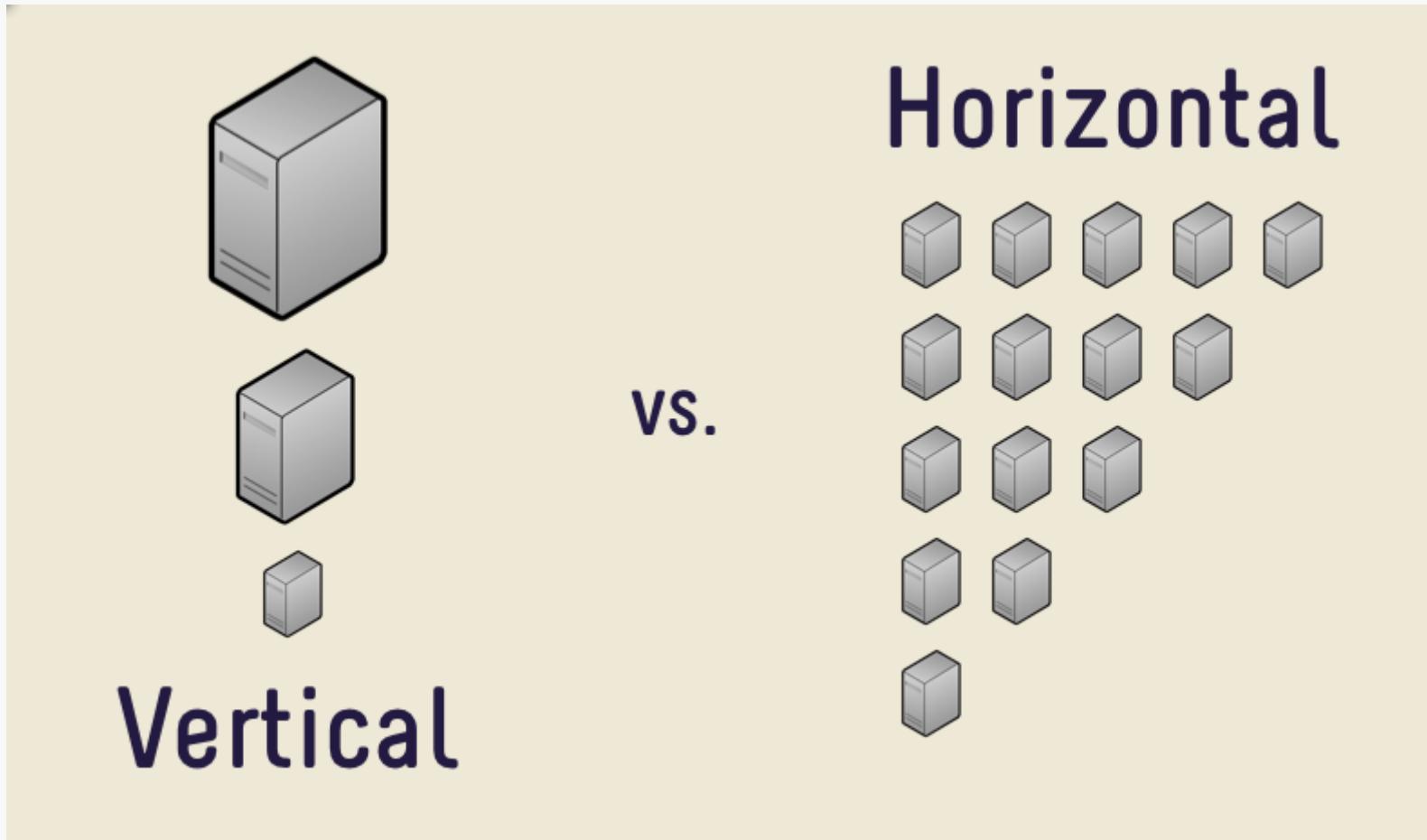


# Por que utilizar containers?

- Isolamento
- Rapidez na criação de instâncias
- Menor dependência o sistema operacional
- Facilidade na criação de ambientes de testes
- Escalabilidade facilitada



# Escalabilidade



# Containers + DevOps

- Facilidades na implementação de automações
- Suporte das principais soluções de mercado como Azure DevOps e GitHub Actions



# Kubernetes: uma visão geral

- Também conhecido como **K8s** ou **kube**
- Desenvolvido originalmente pela **Google**
- Mantido pela **Cloud Native Computing Foundation**
- Escrito em **Go**
- Open source
- Emuladores (**kind**, **Minikube**...)



# Pilares da Observabilidade

- Logs
- Métricas
- Traces (Rastreamento)



# OpenTelemetry: uma visão geral

- Instrumentação em aplicações para coleta de métricas
- Viabiliza a implementação de tracing distribuído de forma descomplicada
- Projeto vinculado à CNCF (Cloud Native Computing Foundation)
- Site: <https://opentelemetry.io/>



# OpenTelemetry: uma visão geral

- Suporte a múltiplas stacks: .NET, Java, Node.js, Python, C++, Go, Erlang/Elixir, PHP, Ruby, Rust, Swift...
- Exporters/soluções de monitoramento com suporte: Console, Grafana Tempo, Jaeger, Zipkin, Elastic APM, Prometheus, Azure Monitor, Application Insights, Dynatrace, AWS CloudWatch, New Relic...

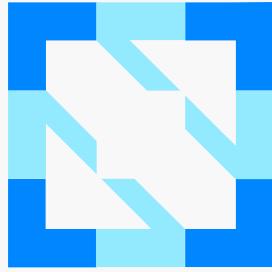


# OpenTelemetry: uma visão geral

- OpenTelemetry Collector: mecanismo para receber, processar e exportar dados de telemetria
- OpenTelemetry Protocol (OTLP) → suporte a HTTP (porta 4318) e gRPC (porta 4317)
- Sampling para coletar apenas percentuais da instrumentação (redução de custos, uso mais racional de recursos computacionais)



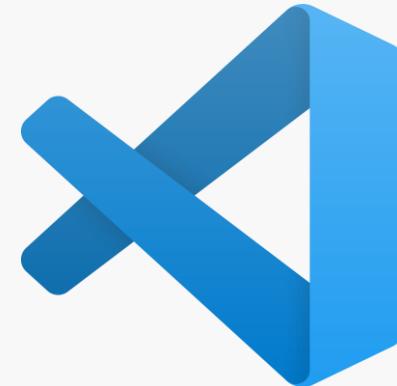
Acompanhe projetos open source...



**CLOUD NATIVE**  
COMPUTING FOUNDATION



**GitHub**



# Twelve-Factor App: uma visão geral

- Concebida dentro da Heroku
- Software as a Service (SaaS)
- Metodologia agnóstica para construção de aplicações robustas
- Princípios com recomendações muito úteis para aplicações em nuvem
- Link: [https://12factor.net/pt\\_br/](https://12factor.net/pt_br/)

# Well-Architected Framework: uma visão geral

- Boas práticas consolidadas no mercado
- Recomendações no uso de recursos
- Governança
- As principais nuvem contam com este mesmo tipo de iniciativa



# Azure Well-Architected Framework: pilares



<https://learn.microsoft.com/en-us/azure/well-architected/>

OBRIGADO!