



# Sistemas Distribuídos

## Agenda

15 minutos

- Daily








45 minutos

- Aula expositiva e dialogada

60 minutos

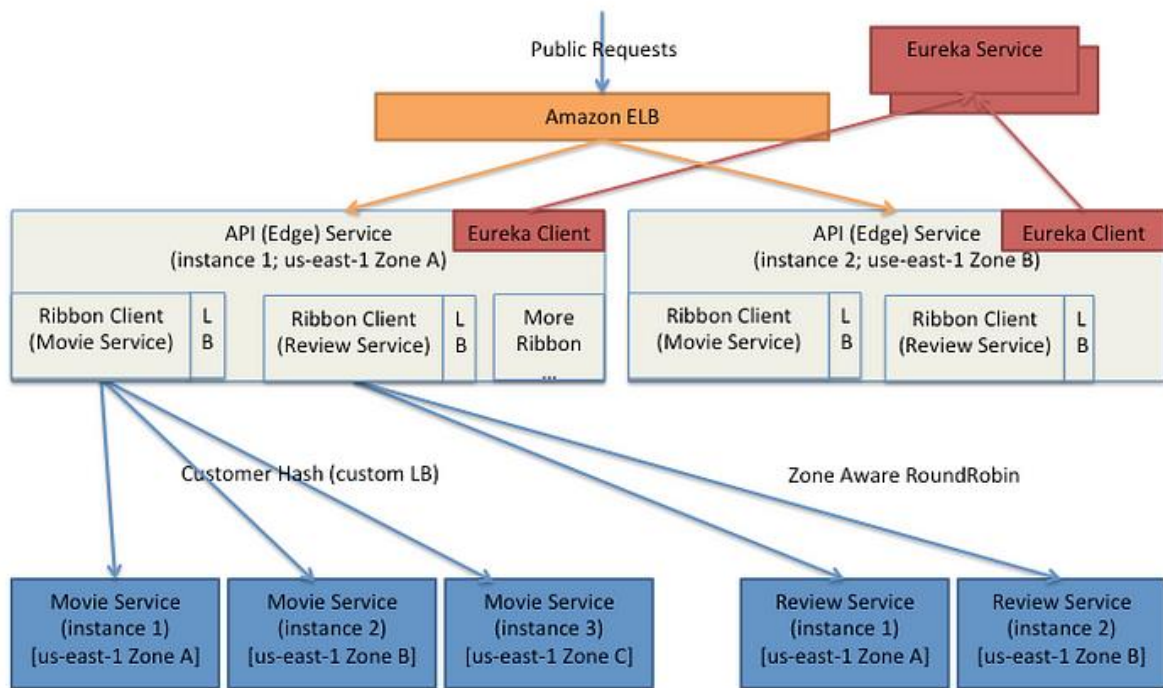
- Ponderada em sala

# Autoestudos

 Sistemas distribuídos	Semana 9	0
 Sistemas distribuídos	Semana 9	0
 Objetos e componentes em sistemas distribuídos	Semana 9	0
 Serviços Web	Semana 9	0
 Sistemas de arquivos distribuídos	Semana 9	0
 Tudo sobre ERP na Nuvem: O que é, Como Funciona e Vantagens	Semana 9	0
 Sistemas Distribuídos	Semana 9	1

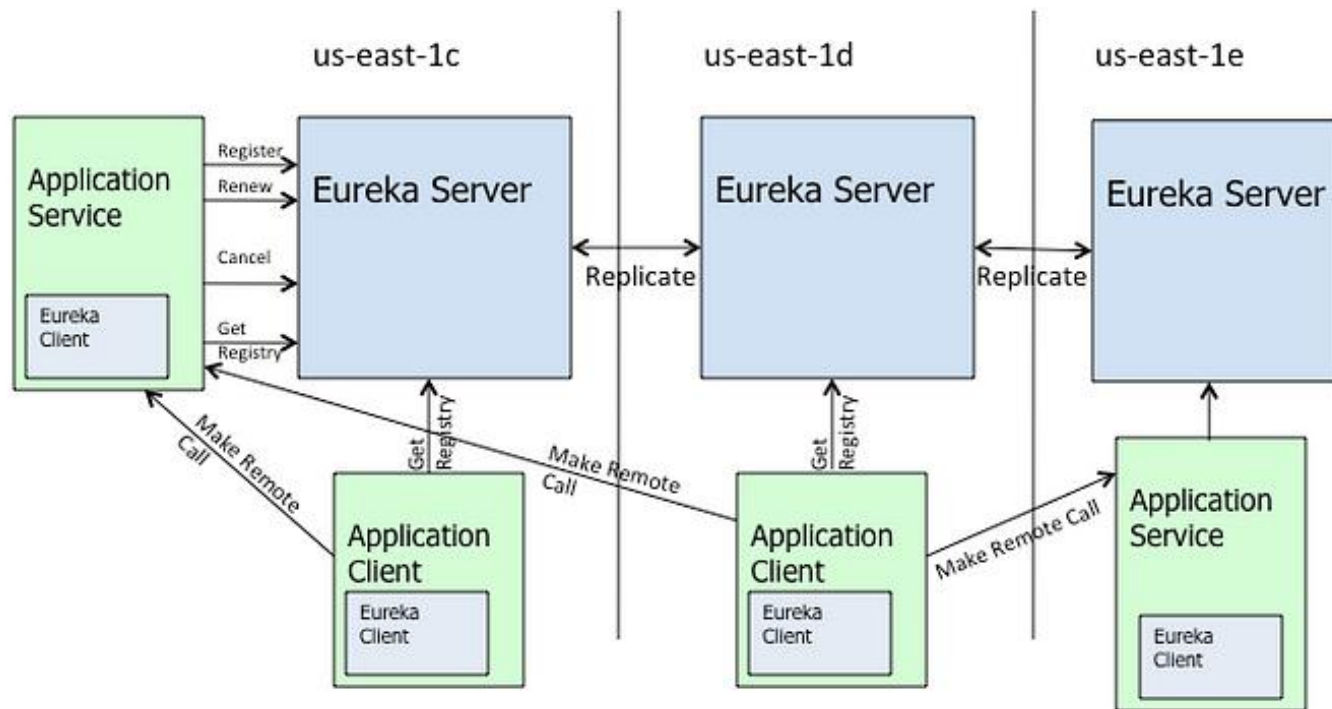


# Topology Netflix Service Oriented Architecture



**Ribbon: client side software load balancing algorithms**

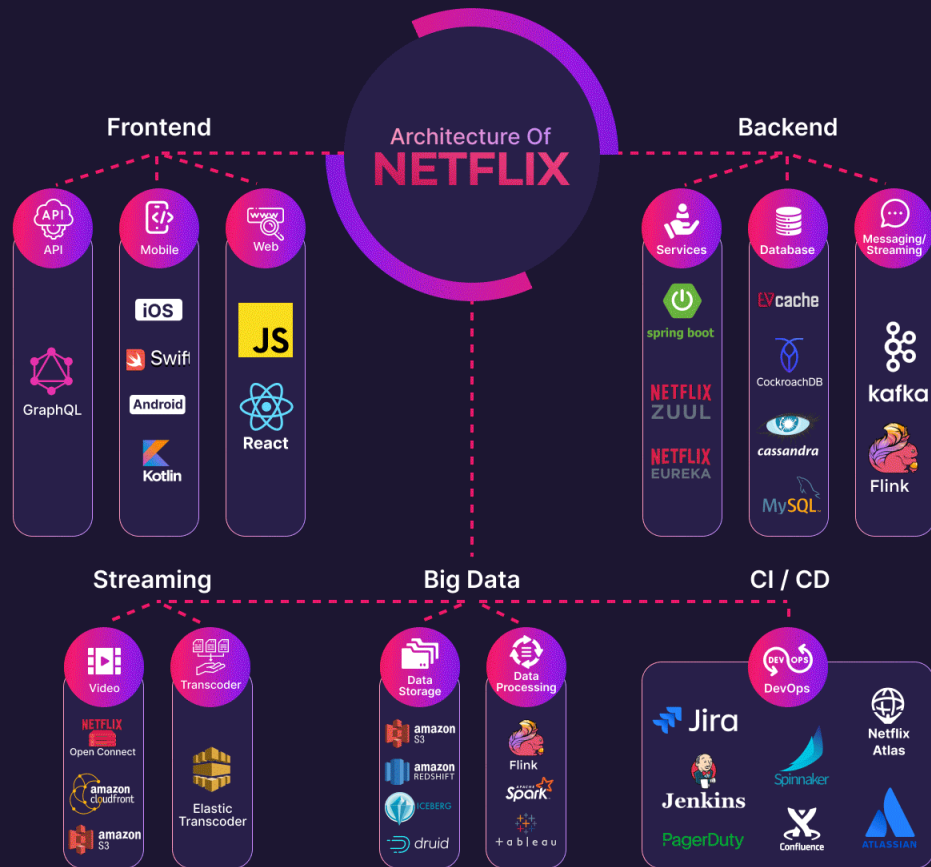
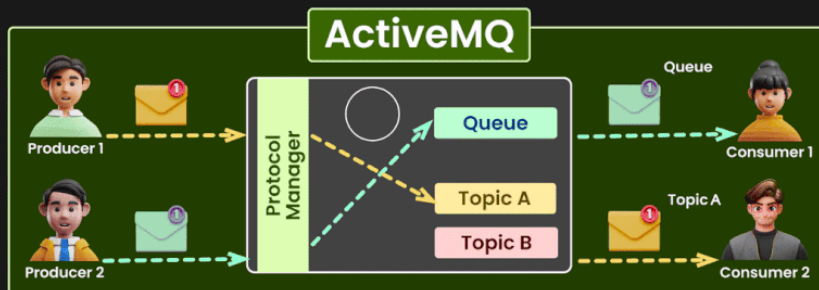
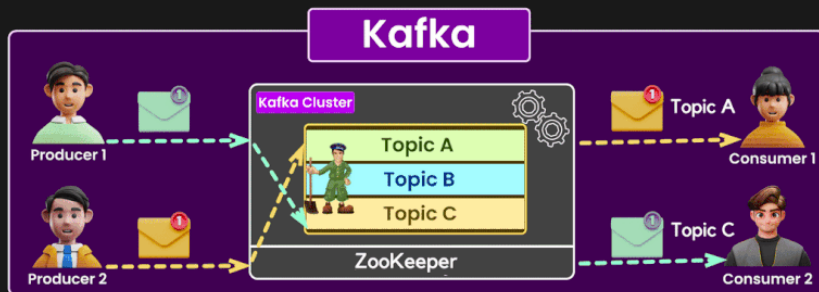
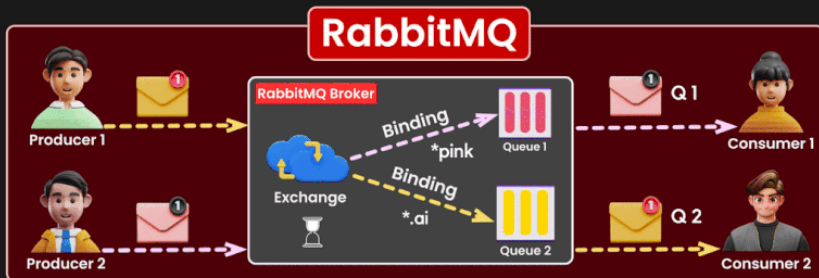
## Eureka High Level Architecture







# RabbitMQ vs Kafka vs ActiveMQ



# Comparativo

HAIL  
O

450+ microservices

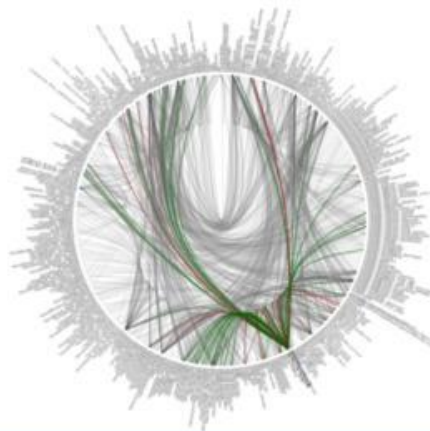


NETFLIX

500+ microservices



500+ microservices





## Internet

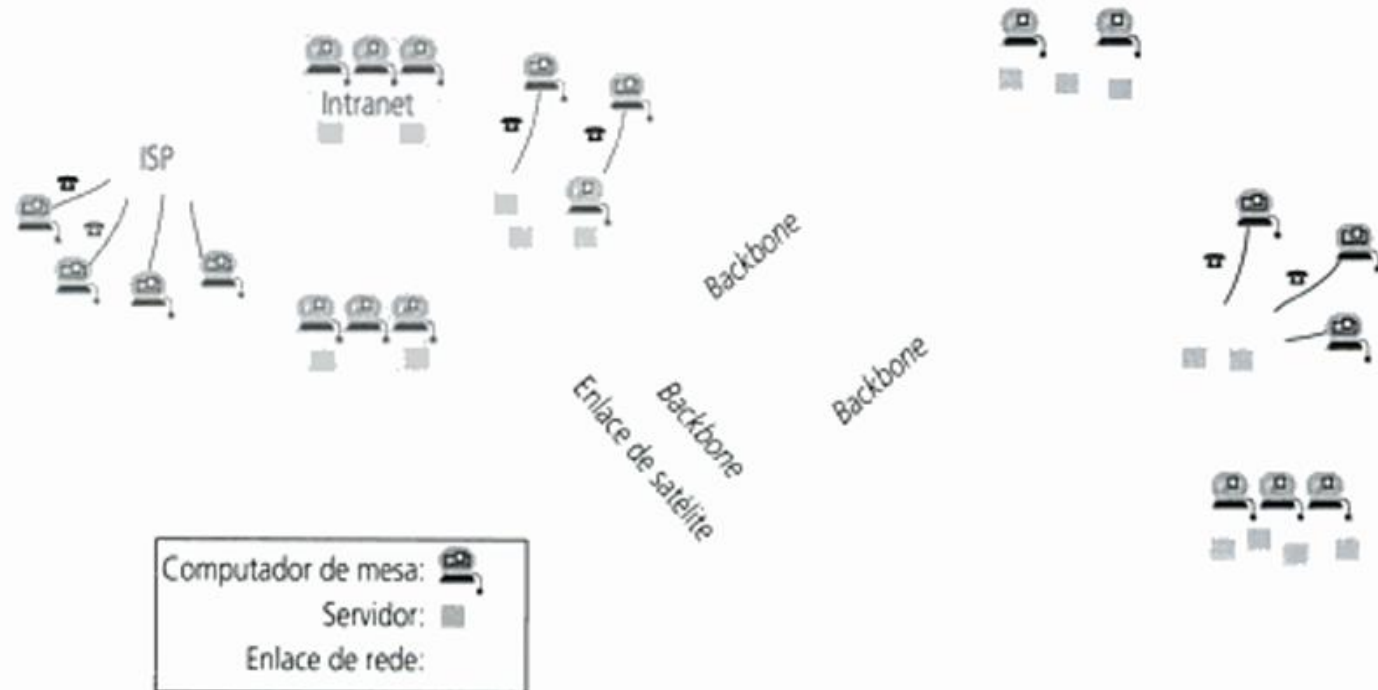
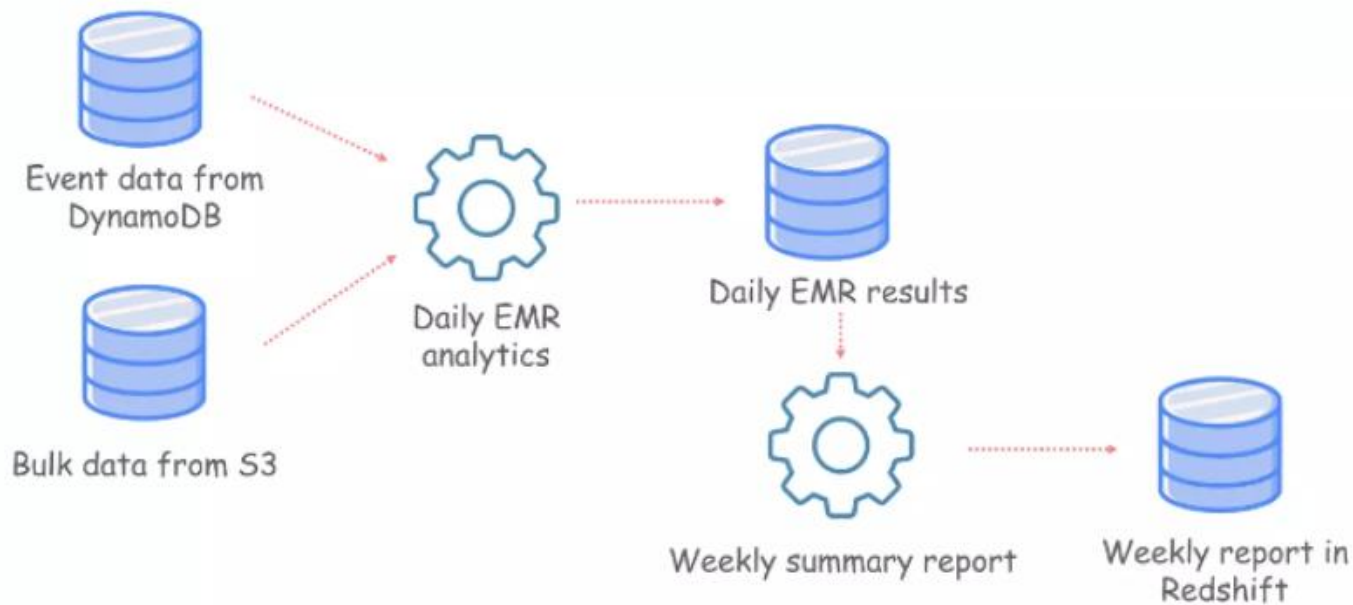


Figura 1.1 Uma parte típica da Internet.

## Big Data com AWS é um Grande Sistema Distribuído



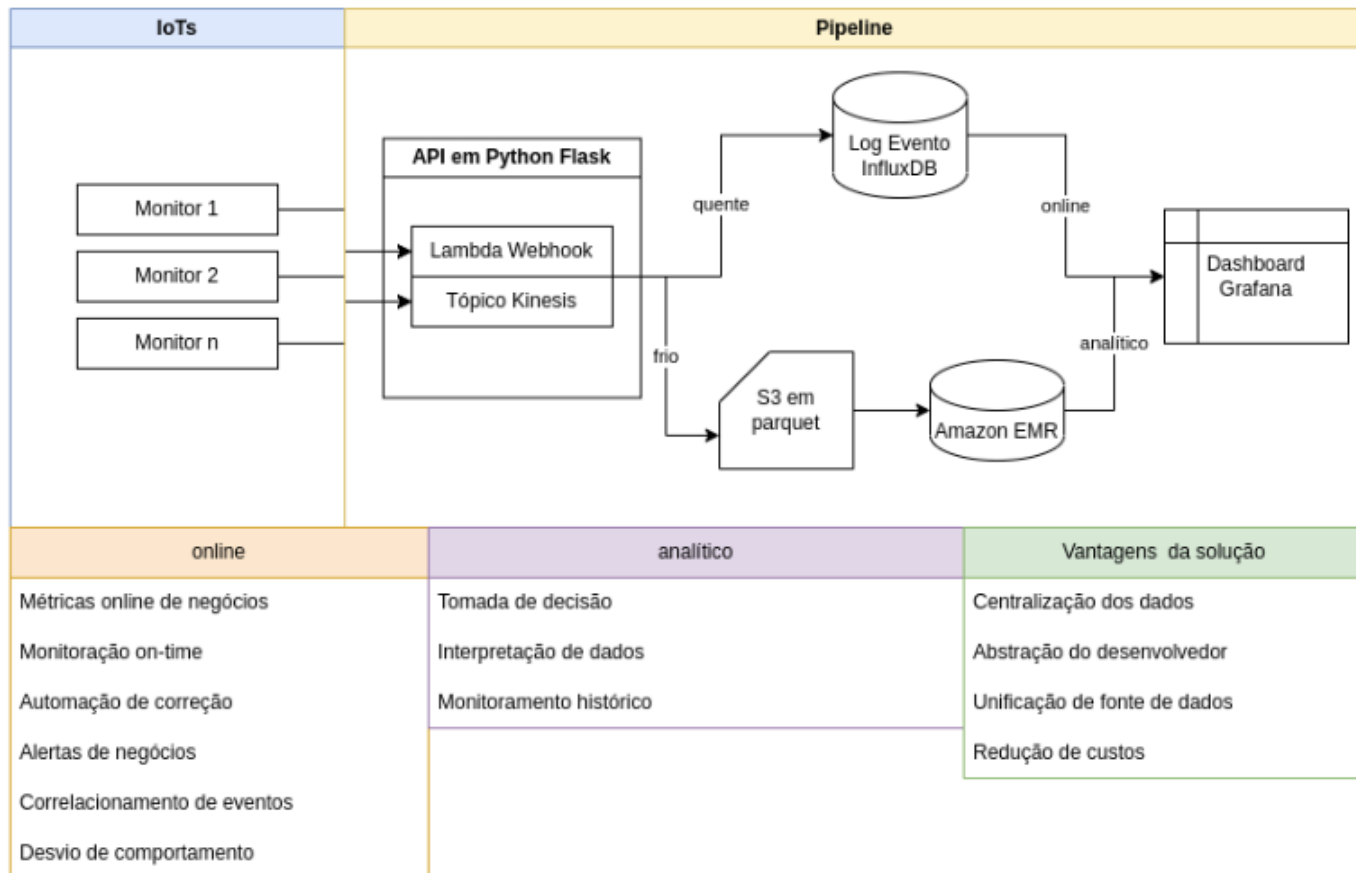
Collect data from different data sources, perform EMR analysis & generate weekly reports



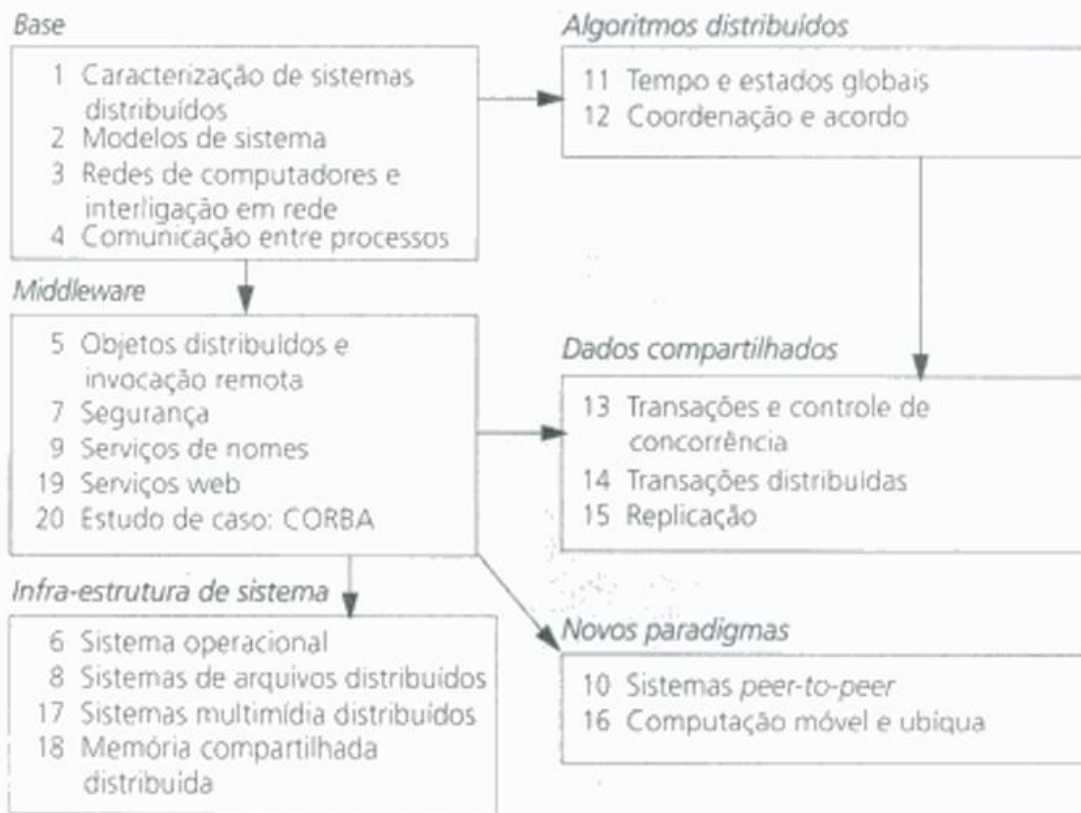
## Problemática: Armazenamento de Logs IoT



# Pipeline de Dados e Métricas



# Sistemas Distribuídos



**CORBA (abreviado de Common Object Request Broker Architecture) é a arquitetura padrão criada pelo Object Management Group para estabelecer e simplificar a troca de dados entre sistemas distribuídos heterogêneos.**

- ORB (Object Request Broker): módulo intermediário entre cliente e objeto, sendo responsável em aceitar a requisição do cliente, enviá-la para o objeto competente e, assim que disponível a resposta, entregá-la para o cliente **(Open Source!)**.
- IDL (Interface Definition Language), uma linguagem baseada em C++ que não possui algoritmos nem variáveis, ou seja, é puramente declarativa, e, portanto, é independente da linguagem de programação utilizada para acessá-la.
- os objetos em sistemas distribuídos possuem uma característica de dualidade: um estado dinâmico, tipicamente alocado em memória volátil (em tempo de execução), e um estado persistente, que não pode ser destruído após o encerramento do programa



## Os Problemas que o Ragnarok teve que enfrentar!

O desenvolvimento de jogos online tem um grande desafio de redes:

- Inconsistência;
- Largura de Banda;
- Latência;
- Perda de Pacotes;

A evolução da infraestrutura ajudou a jogos mais robustos a se popularizarem!



## Uma Breve História do Kubernetes


O Kubernetes começou como um orquestrador de contêineres, aproveitando a popularidade rapidamente adquirida do Docker, a ferramenta de containerização responsável por tornar tudo mais fácil e por esse motivo muito popular, para construir, executar e distribuir contêineres

- O Kubernetes foi criado por Joe Beda, Brendan Burns, e Craig McLuckie, que eram engenheiros do Google, e foi lançado em 2014. O projeto teve origem no Google como um projeto interno chamado "Borg".
- Orquestração de Contêineres;
- Escalabilidade e Alta Disponibilidade;
- Balanceamento de Carga;
- Automação;
- Gestão de Recursos;
- Isolamento e Segurança;
- Portabilidade;

# Exemplo de Arquitetura

Azure Container Registry

Azure Monitor

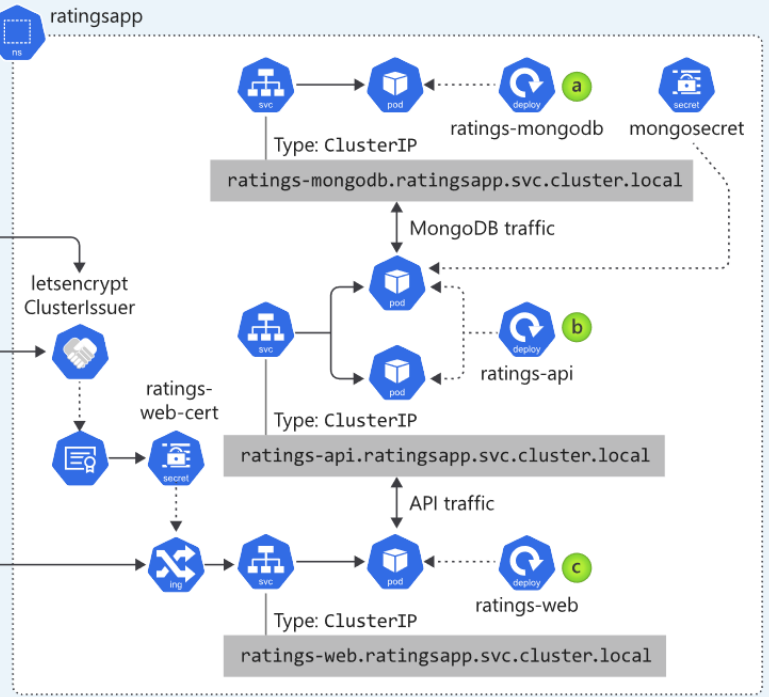
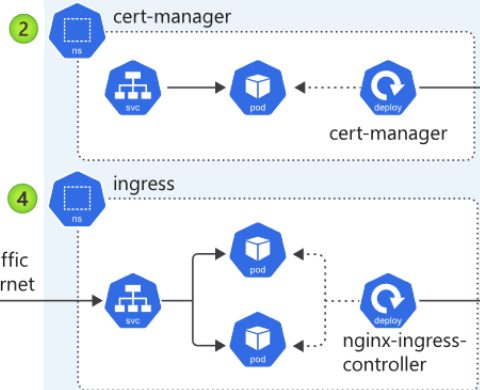
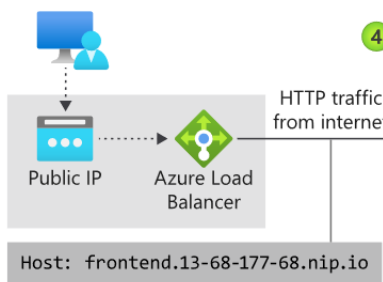
1  Azure Kubernetes Service Engine

Let's Encrypt

2

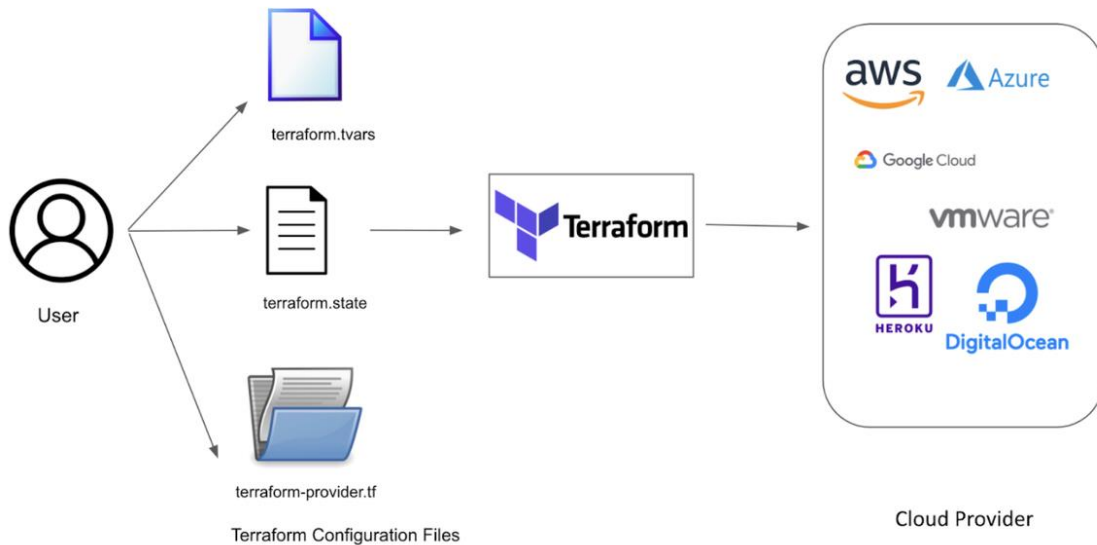
4

3

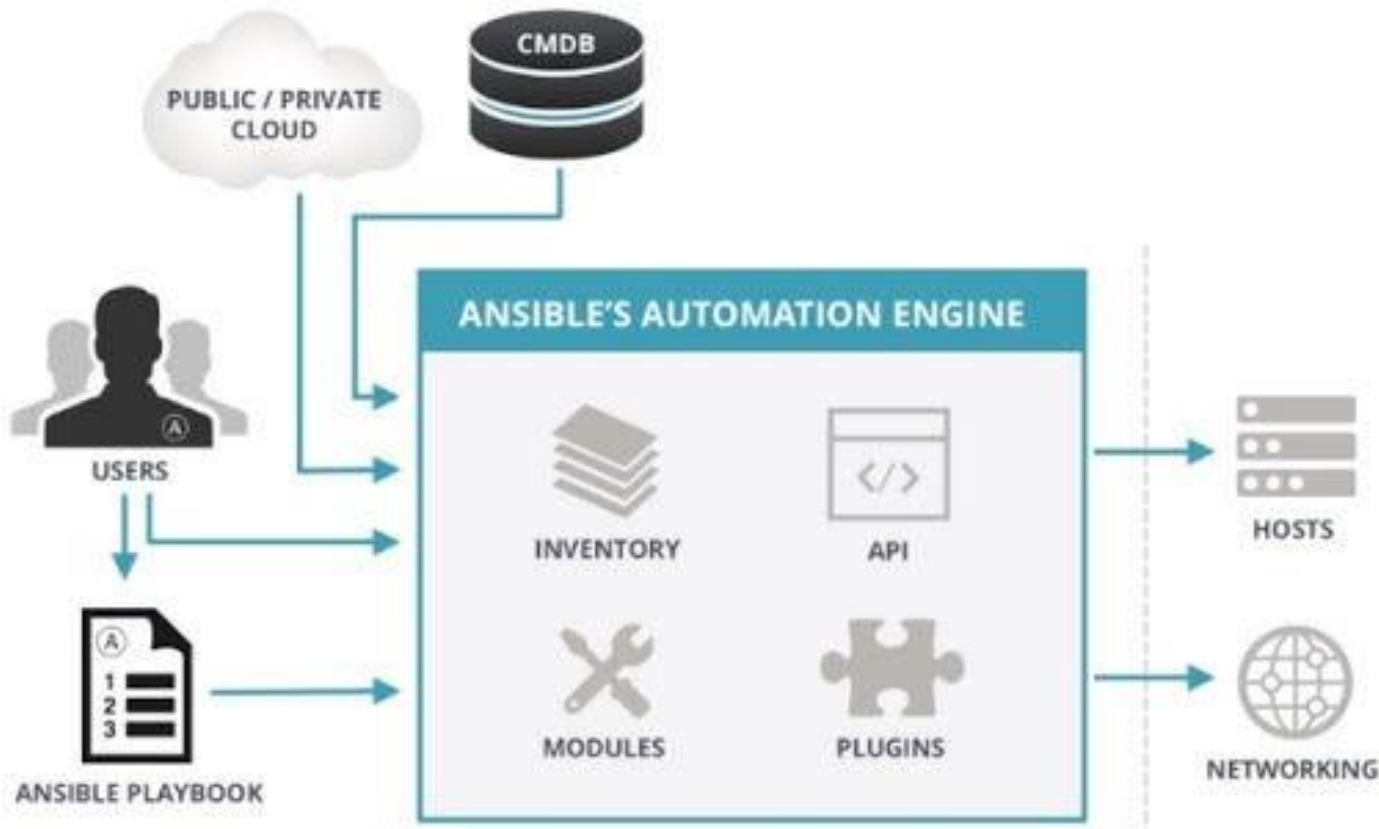


# Terraform e Ansible

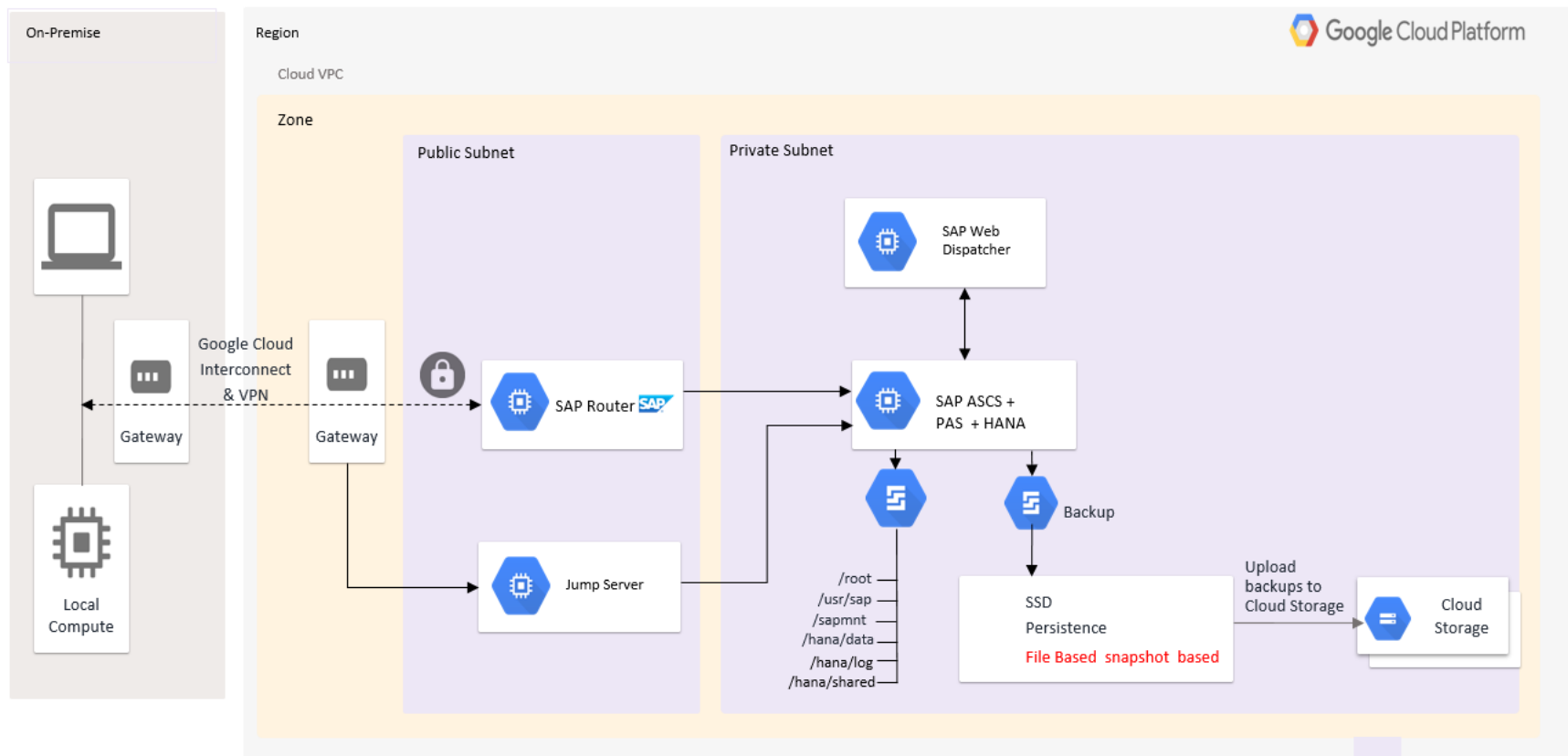
```
1 provider "aws" {
2   region    = "us-east-1"
3   access_key = "${var.access}"
4   secret_key = "${var.secret}"
5 }
6
7 0 references
8 resource "aws_s3_bucket" "btf" {
9   bucket = "meu-bucket-terraform-27042020"
10  acl     = "private"
11
12   tags = {
13     Name           = "My super bucket"
14     Environment    = "Prod"
15   }
16 }
```



# Terraform e Ansible



# SAP HANA no Google? Sim senhor!



<https://cloud.google.com/solutions/sap/docs/architectures/sap-business-suite-on-hana?hl=pt-br>



## Autoestudo Ponderado

Faça uma pesquisa sobre um ERP que seja baseado em uma arquitetura distribuída. Explique seu funcionamento do ponto de vista arquitetural com base nos conceitos que você estudou. Compare com o sistema que está usando em seu projeto, quais são as vantagens e desvantagens?

### Barema

- **Descrição dos elementos da arquitetura do sistema escolhido (tipo de modelo, geração, paradigma de comunicação: 50%;**
- **Comparação com o sistema do projeto: 50%;**

