**Reviews necessários:**

* EC2 purchasing options
* EC2 Contract length
* EC2 instance types
* Difference between load-balancing and auto-scaling
* Number of availability zones in a region
* Module 2 questions after reviewing the things up here

**Termos gerais importantes:**

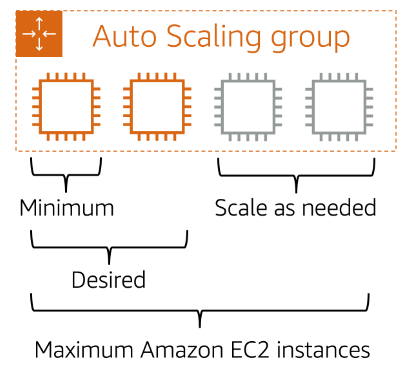
* **On-demand:** Serviços em nuvem disponíveis quando você precisar. Tais serviços não pertencem ao consumidor, e sim ao fornecedor.
* **On-premises:** Serviços físicos, como um servidor físico. O consumidor é responsável por instalar e manter os equipamentos.
* **Up front:** Remete à compra de servidores físicos, por exemplo, onde há a necessidade de realizar a compra, aguardar a chegada. instalar tudo e botar pra rodar
* **Scalability:** É a capacidade do serviço se “moldar” à sua quantidade de demanda. Iniciamos o serviço somente com os recursos necessários, e vamos aumentando-os conforme a demanda aumenta. Podemos escalar **verticalmente** (ou seja, aumentar hardware) ou **horizontalmente** (ou seja, criando novas instâncias de nossas máquinas para atender à demanda adicional).
* **Serverless:** Quando, apesar do código ainda rodar em um servidor, não podemos ver nem acessar a infraestrutura.
* **Cluster:** Uma série de instâncias EC2 (contêineres) rodando juntos.
* **Multitenancy:** Compartilhamento de recursos (hardware) entre VMs (instâncias EC2). O **Hypervisor** da AWS coordena o Multitenancy.
* **CaaS:** Compute as a Service model

**Amazon EC2:**

**Termos importantes:**

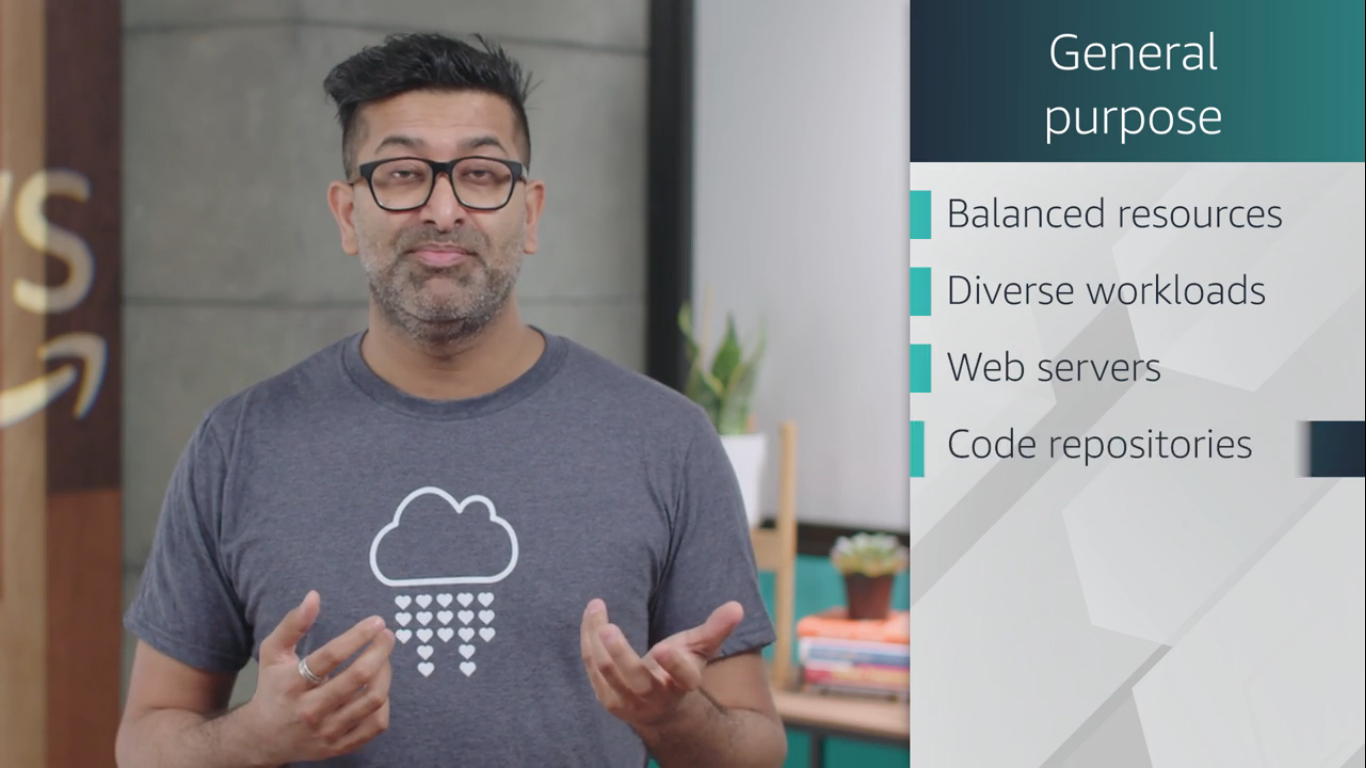
* **Amazon EC2 Auto-Scaling:** Utilizado para escalar as instâncias de EC2 horizontalmente. Podemos utilizar o **dynamic scaling**, que responde dinamicamente às novas demandas, ou o **predictive scaling**, que agenda um número de instâncias a serem upadas em determinados horários, baseado em previsão.

Ao iniciar um **Auto-Scaling Group**, definimos o número mínimo, o número “desejado” e o número máximo de instâncias. Só pagamos pelas instâncias que usamos, quando usamos.

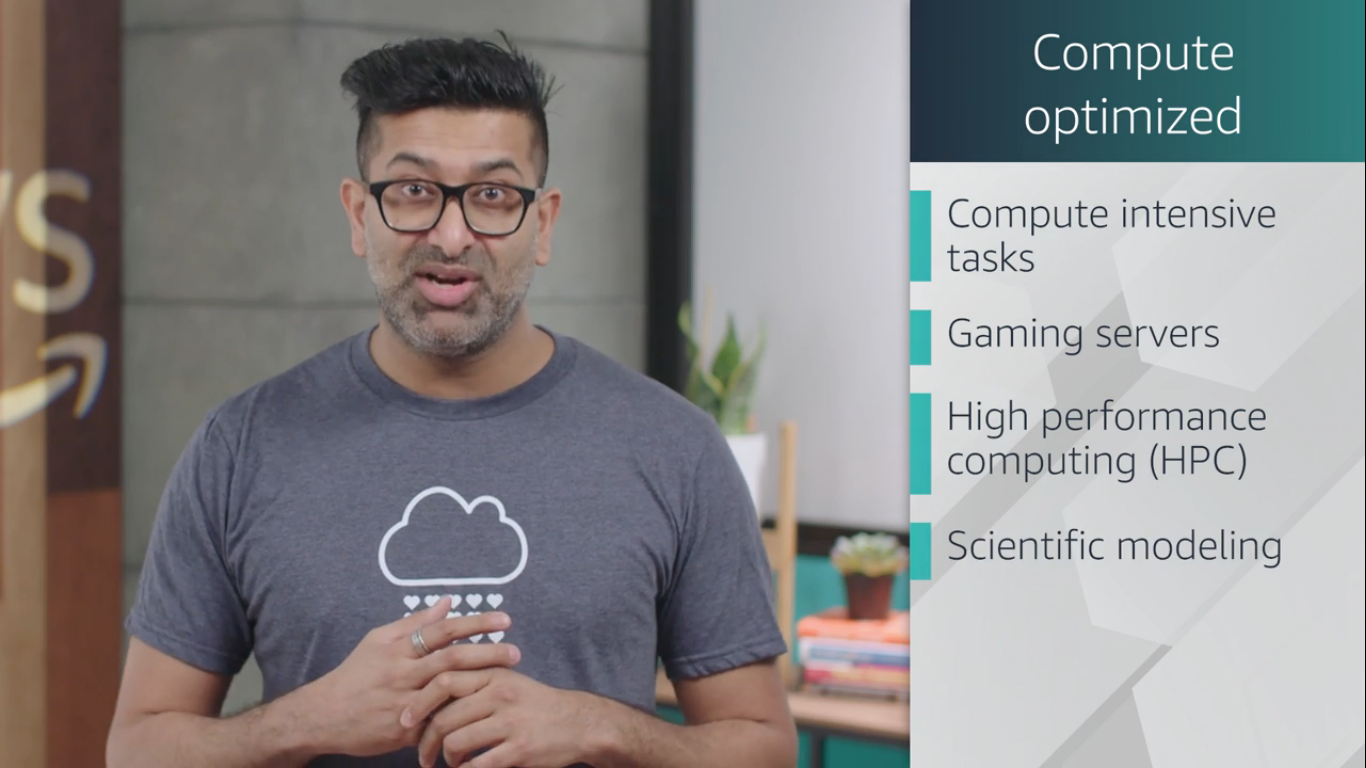
****

**Tipos de instâncias EC2:**

**General purpose:**



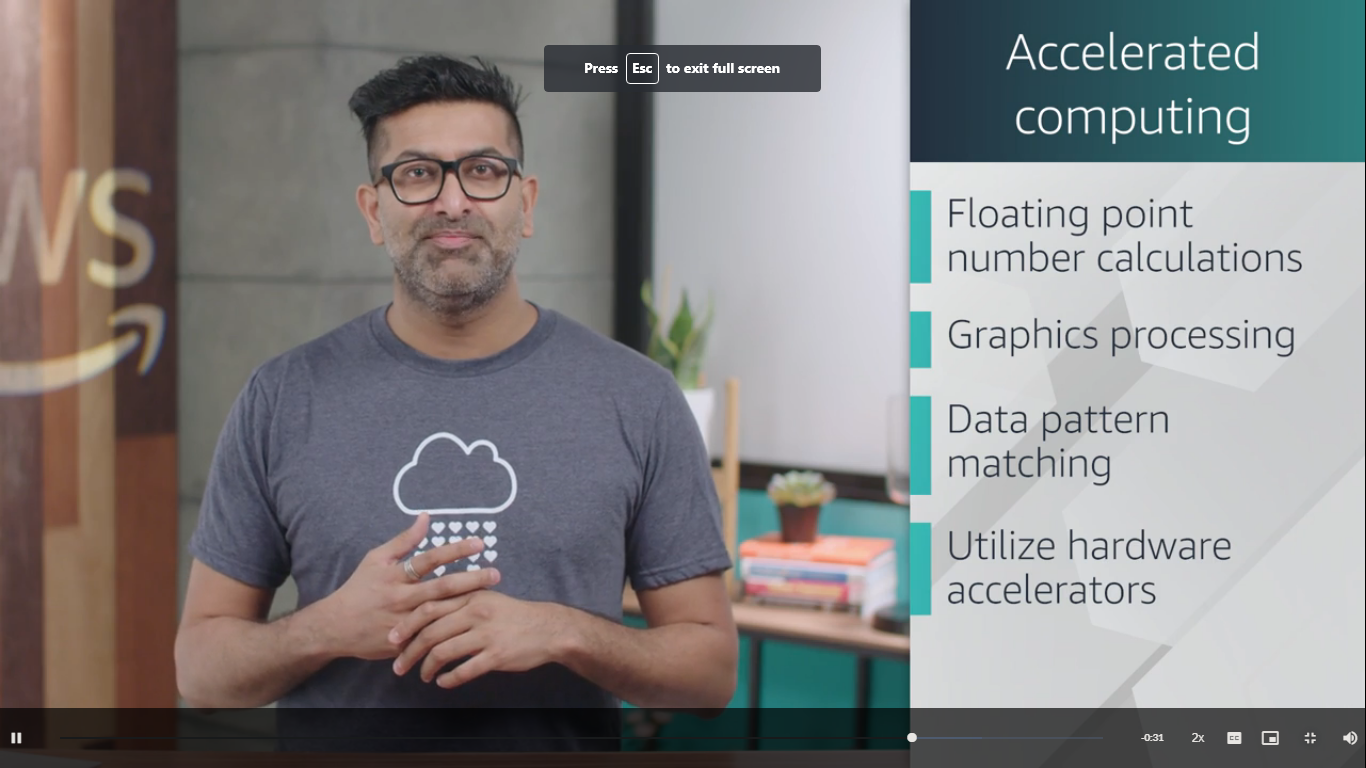
**Compute optimized:**



**Memory optimized:**



**Accelerated computing:**



**Storage optimized:**



**Tipos de billing EC2:**

**On-demand:** Você será cobrado pelo tempo que sua instância rodou, podendo ser por hora ou segundo, dependendo do tipo do SO escolhido. Não são necessários contratos nem avisos para com a AWS.

**Savings plan:** Oferece baixos preços no uso do EC2, em troca de um uso consistente, com contrato de 1 ou 3 anos. Pode oferecer descontos de até 72%. Também se aplica para FARGATE e AWS LAMBDA.

**Reserved instances:** Parauso previsível. Contrato de 1 a 3 anos. 3 métodos de pagamento disponíveis: **up front** (pagamento total antecipado no contrato), **partial up front** (pagamento parcial no contrato) e **no up front** (quando nada é pago no início do contrato).

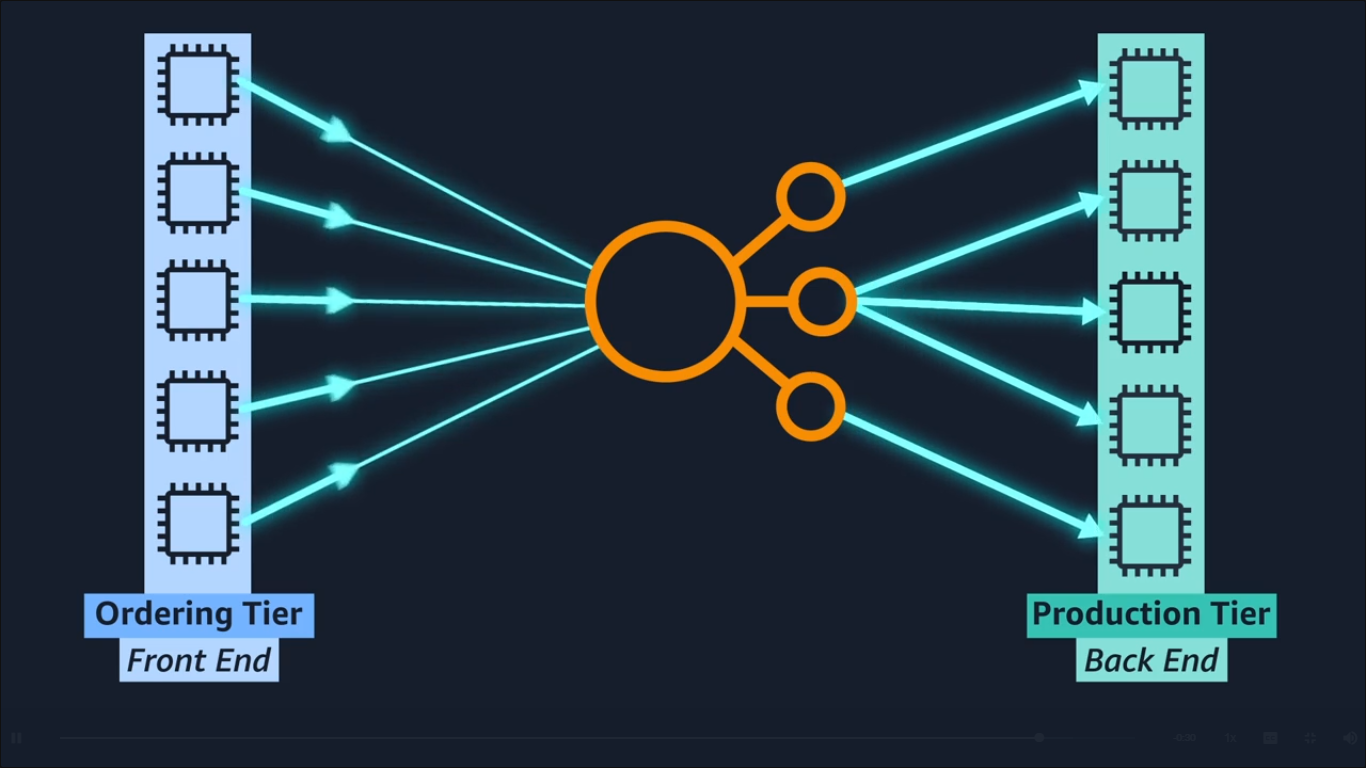
**Spot instances:** São instâncias “emprestadas”, que podem ser reinvindicadas pela AWS a qualquer momento, ou seja, podem ser interrompidas. Têm um desconto absurdamente alto, mas possuem esse drawback de serem interrompidas. Ideais para batch jobs.

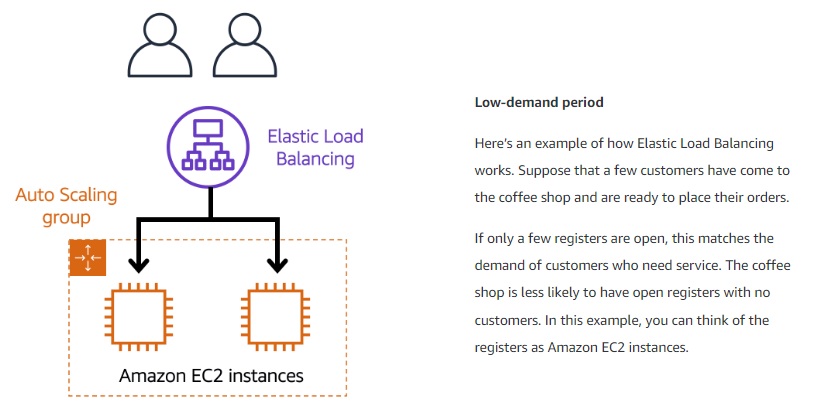
**Dedicated hosts:** Servidores físicos. Seus recursos não possuem multitenancy, ou seja, não são compartilhados com outras VMs. Ideiais quando a empresa precisa atingir requisitor de compliance.

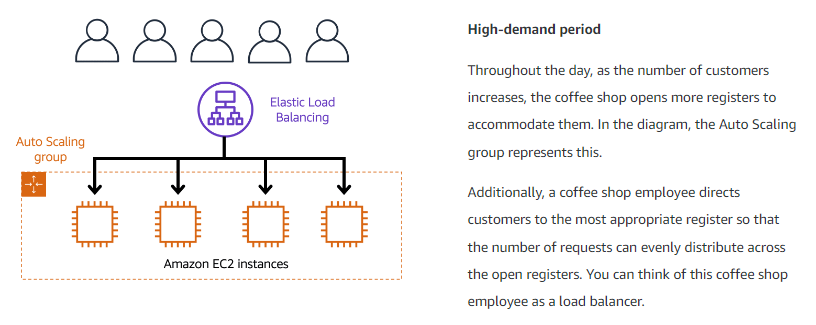
**Amazon ELB (Elastic Load Balancing)**

**Utilidade geral:**

* Serve para direcionar o tráfego entre as instâncias de EC2, por exemplo. Sua finalidade é não permitir que uma instância seja overloaded e outra underloaded.



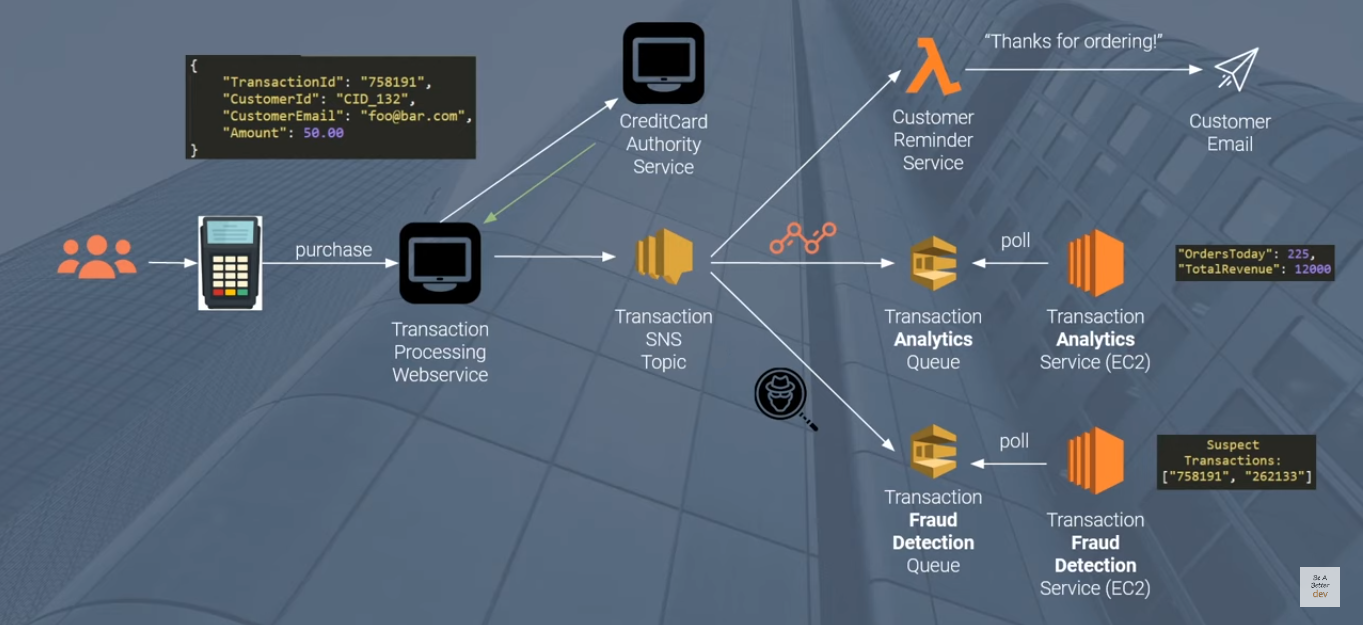
****

****

**Amazon SNS (Simple Notification Service)**

**Utilidade geral:**

* Serviço de publish/subscribe. Sistemas producers podem publicar mensagens para sistemas consumers (como o SQS, por exemplo), através do SNS Topic, que é responsável por realizar o fanout (direcionar a mesma mensagem para diversos consumers). Abaixo, segue uma boa ilustração de seu use-case ([SNS vs SQS Comparison? Whats the difference? | Learn with a practical example - YouTube](https://www.youtube.com/watch?v=mXk0MNjlO7A)):



**Amazon SQS (Simple Queue Service)**

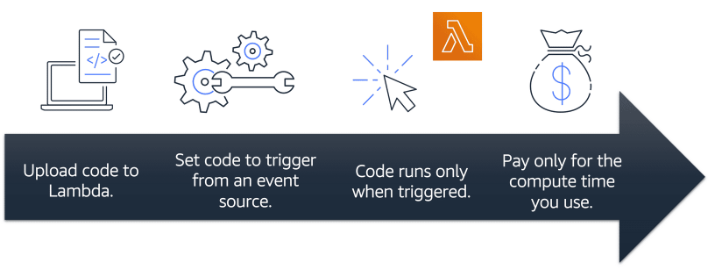
**Utilidade geral:**

* Uma fila, onde mensagens são publicadas e consumidas. Para melhor entendimento, consultar explicação do SNS acima.

**AWS Lambda**

**Utilidade geral:**

* Serviço onde subimos um código em uma **lambda function**, que será rodada de forma Serverless em um ambiente escalável e gerenciado pela AWS após a ativação de um trigger configurado. Designado para processamento rápido, que levam menos que 15 minutos. Não recomendado para deep-learning, por exemplo.





**Amazon ECS (Elastic Container Service)**

**Utilidade geral:**

* Serve para gerenciar aplicações contêinerizadas de forma escalável, sem que precisemos lidar com algum software próprio de orquestração de contêineres.

**Amazon EKS (Elastic Kubernetes Service)**

**Utilidade geral:**

**AWS FARGATE**

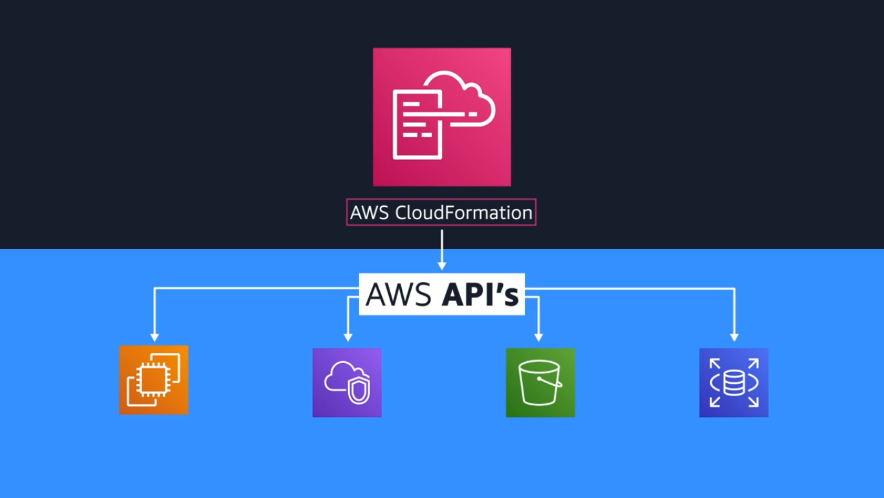
**Utilidade geral:**

* [**AWS Fargate**](https://aws.amazon.com/fargate/)is a serverless compute engine for containers. It works with both Amazon ECS and Amazon EKS. When using AWS Fargate, you do not need to provision or manage servers. AWS Fargate manages your server infrastructure for you.

**AWS CloudFormation**

**Utilidade geral:**

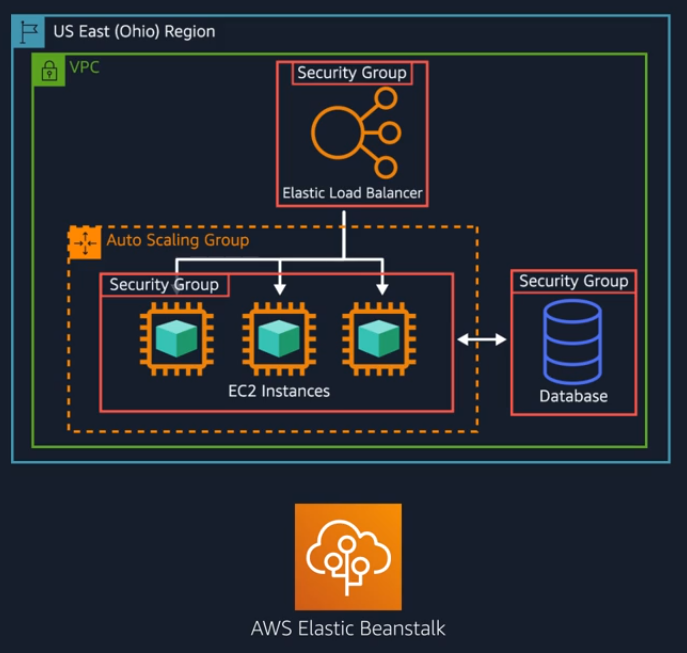
* [**AWS**](https://aws.amazon.com/fargate/) **CloudFormation**é uma ferramenta que permite a geração de ambientes através de um arquivo .json ou .yaml. Ele gerencia a chamada às APIs através de uma estrutura de documento definida.



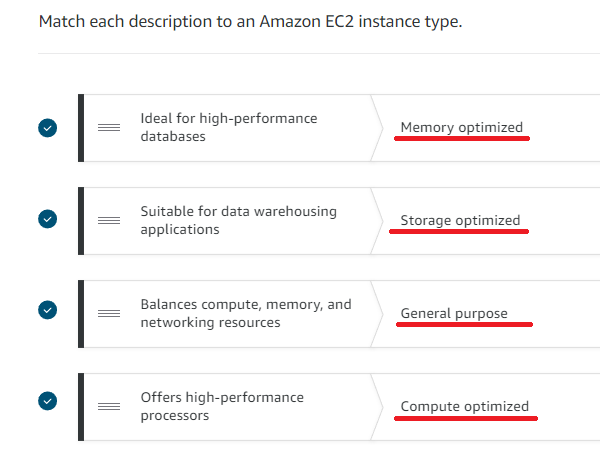
**AWS Elastic Beanstalk**

**Utilidade geral:**

* **AWS Elastic Beanstalk** é uma ferramenta que gera um ambiente a partir do código e configurações informadas. Com ele, também a possível salvar as configurações do ambiente para futuros deploys

****

**Tipos de instâncias EC2 (Lesson 7)**



**Tipos de pagamentos EC2 (Lesson 8)**

****

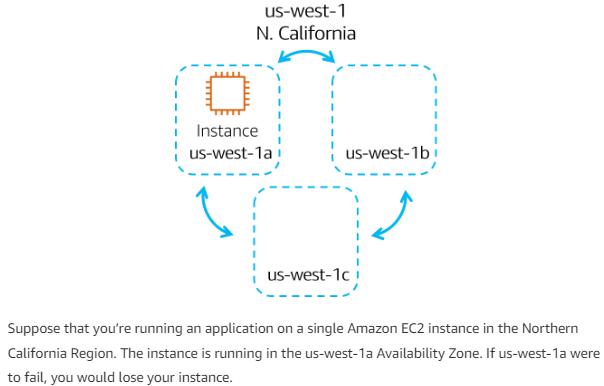
**Regions**

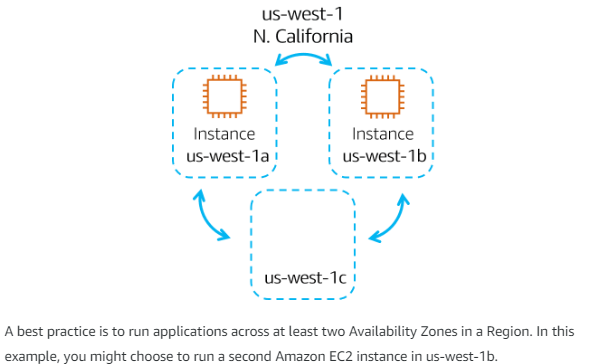
**Key factors to choose a region: (IMPORTANT! Module 3 Class 2)**

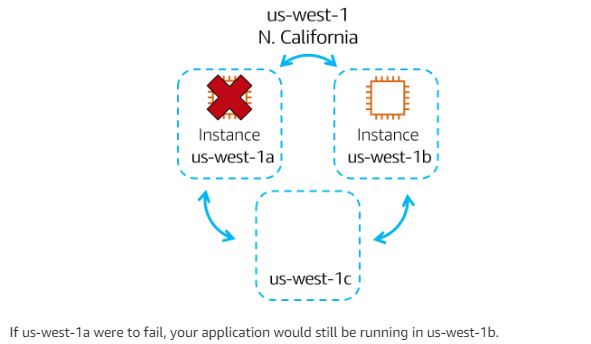
* Compliance
* Proximity
* Feature availability
* Pricing

**Regionally scoped service: ??????????? PESQUISAR!**

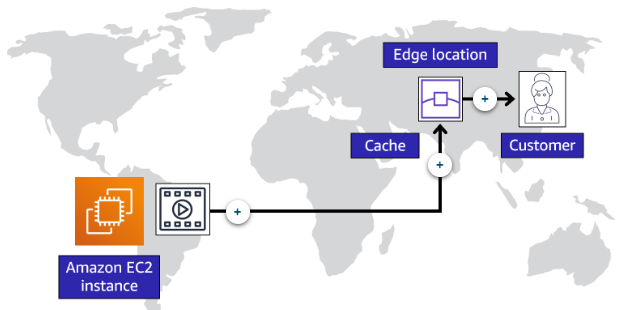
**Availability zones:** Each **region** has **TWO OR MORE availability zones**. They are physical buildings spread across a region.



****

****

**Edge locations and Amazon CloudFront (is a CDN): CloudFront** is usedwhen a client from a far region needs to access data from another region. Brazil being the origin and China the destiny, for example, we can use CloudFront to set a copied cached version of Brazil’s product into a chinese region (called edge location), reducing latency when a chinese client makes a request to that product.



**AWS Outposts:** Physical environment provided by AWS when requested by a client.

**Route 53:** Edge locations run **Amazon Route 53**, which help to direct customers to the correct web location.