

# Cloud Computing - AWS



Presentado por **Alejandro Chacón**

[www.consultec-ti.com](http://www.consultec-ti.com)

# Agenda



Intro



Almacenamiento S3



Networking



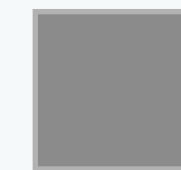
Cómputo y Almacenamiento



Integración y Monitoreo



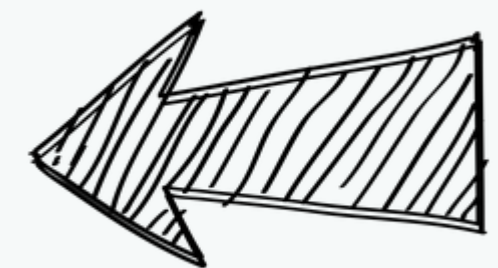
Bases de Datos



Seguridad



Despliegues y Gestión de Infraestructura

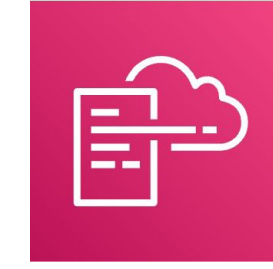


# DevOps Tools AWS

- Intro
- Cloudformation
- Cloud Development Kit – CDK
- Beanstalk
- Code
  - {Build/Pipeline/Deploy/Commit/Star/Artifact}
- Cloud9
- System Manager SSM
- OpsWorks

# DevOps Tools AWS

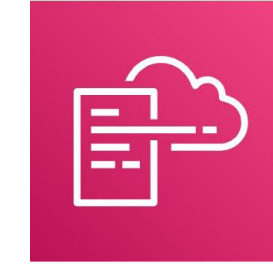
## CloudFormation



- CloudFormation es una forma declarativa de describir infraestructura de AWS, para cualquier recurso (la mayoría de ellos son compatibles).
- Por ejemplo, dentro de una plantilla de CloudFormation, podemos configurar:
  - un grupo de seguridad
  - instancias EC2 que usen un específico SG
  - Buckets S3
  - Un ALB frente a estas EC2
- Luego, CloudFormation los crea usando los archivos de configuración de Infraestructura como código, en el orden correcto, con la configuración específica descrita.

# DevOps Tools AWS

## CloudFormation – Beneficios

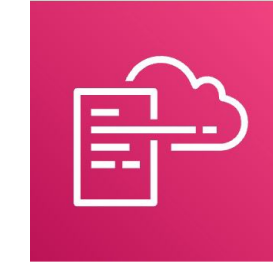


- Infraestructura como código
  - No se crean recursos manualmente, lo cual es excelente para el control
  - Los cambios en la infraestructura se revisan a través del código.
- Costo
  - Cada recurso dentro de un stack que está etiquetado con un tag para que pueda ver fácilmente en el cost manager.
  - Puede estimar los costos de sus recursos usando la plantilla de CloudFormation
  - Estrategia de ahorro: en Dev, podría automatizar la eliminación de plantillas a las 5 p. m. y volver a crearlas a las 8 a. m., de manera segura



# DevOps Tools AWS

## CloudFormation – Beneficios



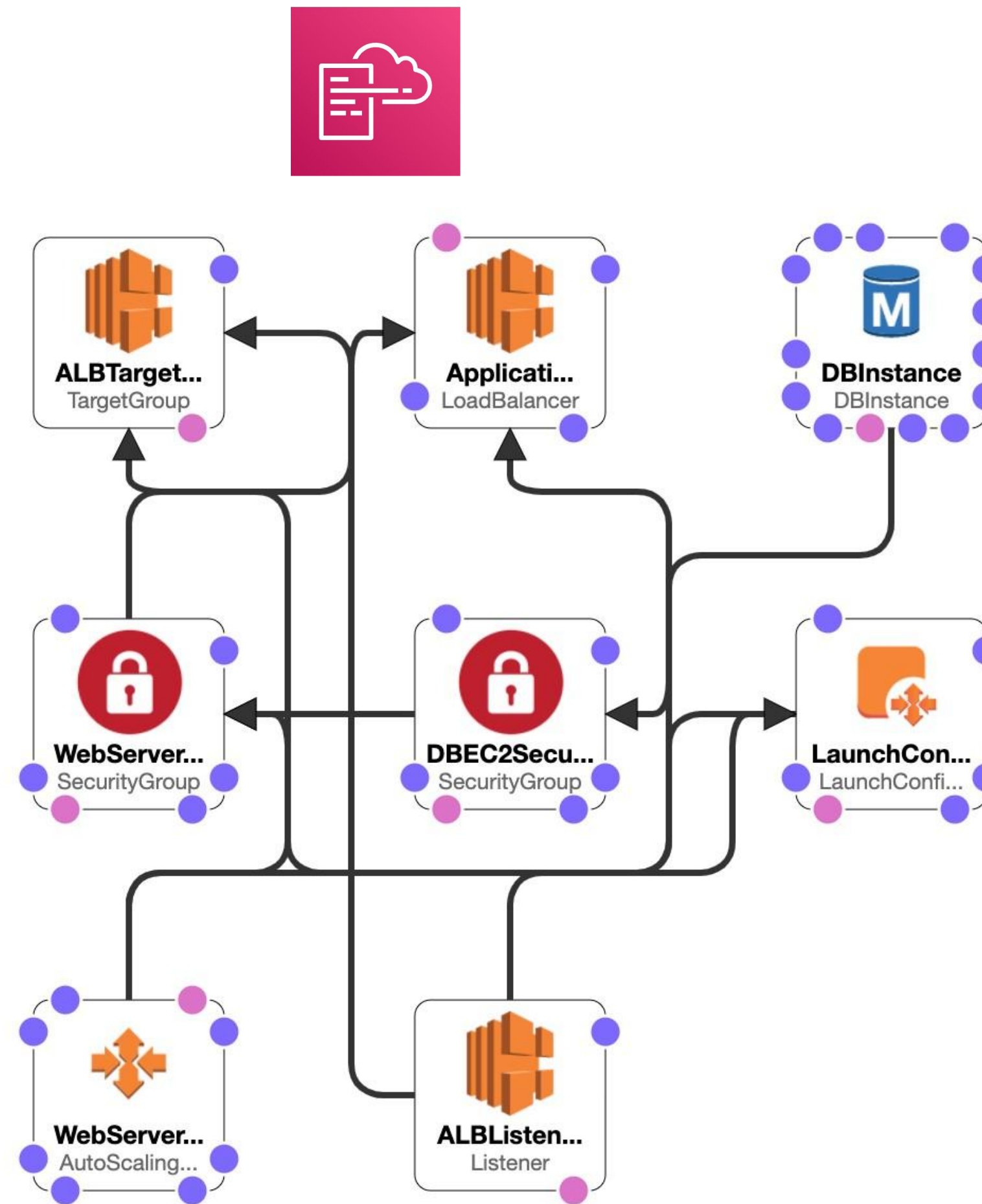
- Productividad
  - Capacidad para destruir y volver a crear una infraestructura en la nube sobre la marcha
  - ¡Generación automática de diagramas para sus plantillas!
  - Programación declarativa (no es necesario averiguar el orden y la orquestación)
- No reinventes la rueda
  - ¡Aproveche las plantillas existentes en la web!
  - Aprovecha la documentación
- Admite (casi) todos los recursos de AWS:
  - Todo lo que veremos en este curso es compatible.
  - Puede usar "recursos personalizados" para los recursos que no son compatibles

# DevOps Tools AWS

## CloudFormation – Stack Builder

Ejemplo: Pila de WordPress  
CloudFormation

Podemos ver todos los recursos  
Podemos ver las relaciones  
entre los componentes

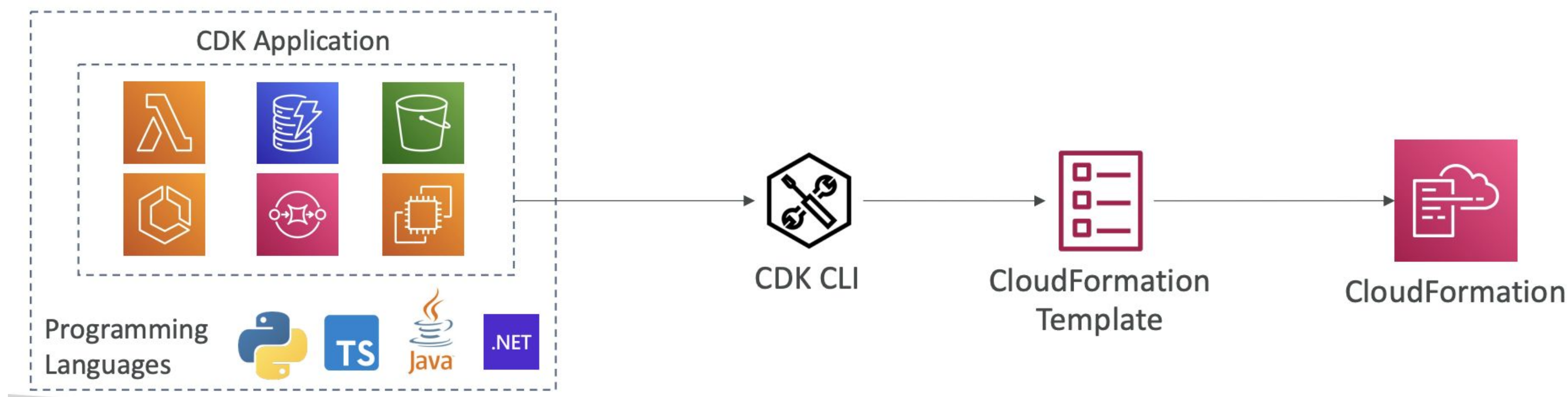


# DevOps Tools AWS

## CDK – Kit de Desarrollo



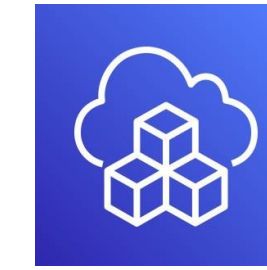
- Define infraestructura en la nube usando un lenguaje familiar:
  - JavaScript/TypeScript, Python, Java y .NET
  - El código se "compila" en una plantilla de CloudFormation (JSON/YAML)
- Por lo tanto, puede implementar la infraestructura y el código de tiempo de ejecución de la aplicación juntos.
  - Genial para funciones Lambda
  - Ideal para contenedores Docker en ECS/EKS





# DevOps Tools AWS

## CDK – Kit de Desarrollo



```
export class MyEcsConstructStack extends core.Stack {
  constructor(scope: core.App, id: string, props?: core.StackProps) {
    super(scope, id, props);

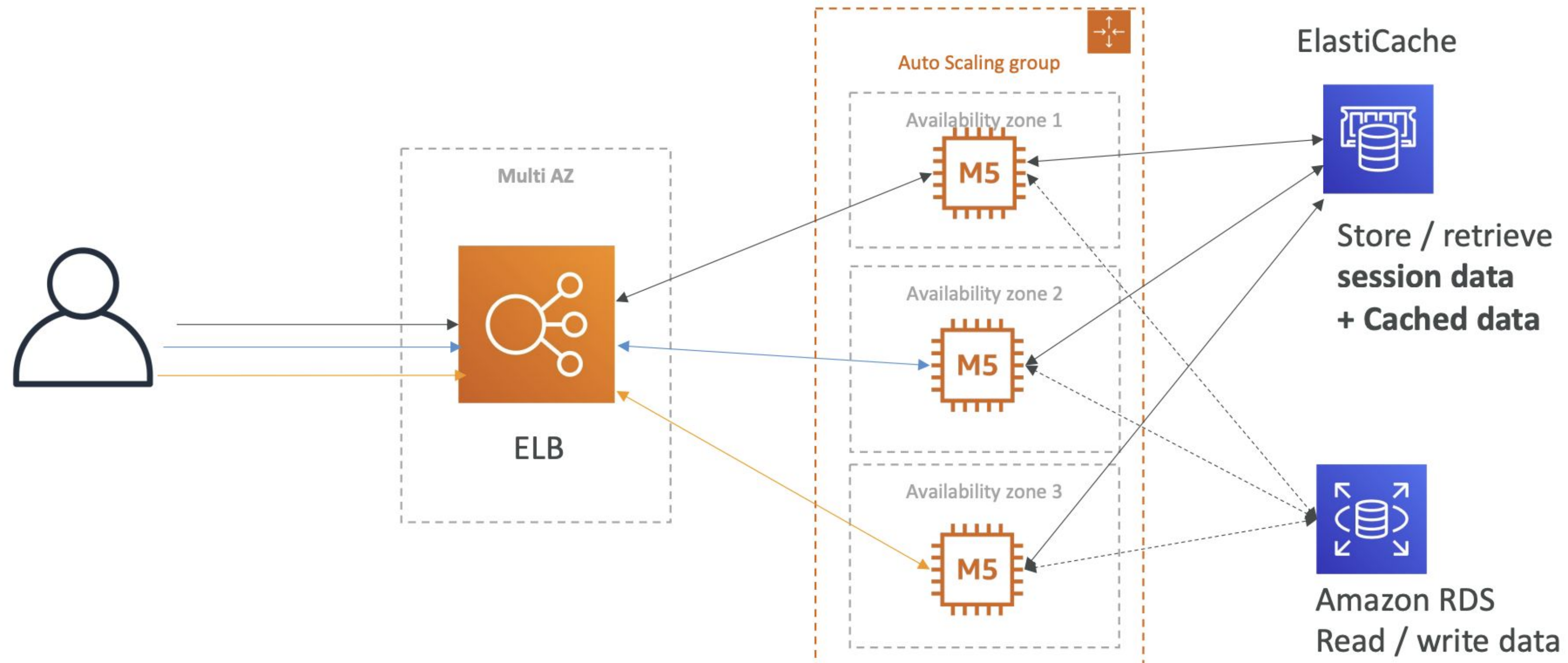
    const vpc = new ec2.Vpc(this, "MyVpc", {
      maxAzs: 3 // Default is all AZs in region
    });

    const cluster = new ecs.Cluster(this, "MyCluster", {
      vpc: vpc
    });

    // Create a load-balanced Fargate service and make it public
    new ecs_patterns.ApplicationLoadBalancedFargateService(this, "My
      cluster: cluster, // Required
      cpu: 512, // Default is 256
      desiredCount: 6, // Default is 1
      taskImageOptions: { image: ecs.ContainerImage.fromRegistry("an
      memoryLimitMiB: 2048, // Default is 512
      publicLoadBalancer: true // Default is false
    });
  }
}
```

# DevOps Tools AWS

## Arquitectura Ejemplo - 3 Tiers



# DevOps Tools AWS

## Algunos desafíos que los Devs enfrentan:

- Gestión de infraestructura
- Despliegues del Código
- Configuración de todas las bases de datos, balanceadores de carga, etc.
- Preocupaciones de escala
- La mayoría de las aplicaciones web tienen la misma arquitectura
  - (ALB + ASG)
- ¡Todo lo que los desarrolladores quieren es que su código se ejecute!
- Consistentemente a través de diferentes aplicaciones y entornos



# DevOps Tools AWS

## Elastic Beanstalk – Intro

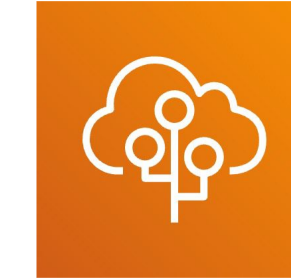


- Elastic Beanstalk ayuda a enfocar al desarrollador de la implementación de una aplicación en AWS
  - Utiliza todos los componentes que hemos visto antes:
    - EC2, ASG, ELB, RDS, etc...
  - ¡Todo en una pantalla fácil de entender!
  - Tenemos control total sobre la configuración.
- 
- Beanstalk = Plataforma como servicio (PaaS)
  - Beanstalk es gratis pero pagas por las instancias utilizadas.



# DevOps Tools AWS

## Elastic Beanstalk – Intro



- Servicio gestionado
  - Beanstalk maneja la configuración de la instancia/SO
  - La estrategia de implementación es configurable pero la realiza Elastic Beanstalk
  - Aprovisionamiento de capacidad
  - Equilibrio de carga y escalado automático
  - Supervisión del estado de la aplicación y capacidad de respuesta
- Solo el código de la aplicación es responsabilidad del desarrollador.
- Tres modelos de arquitectura:
  - Implementación de instancia única: bueno para el desarrollo
  - LB + ASG: excelente para aplicaciones web de producción o preproducción
  - Solo ASG: ideal para non-web apps en producción (workers, etc.)

# DevOps Tools AWS

## Elastic Beanstalk – Intro



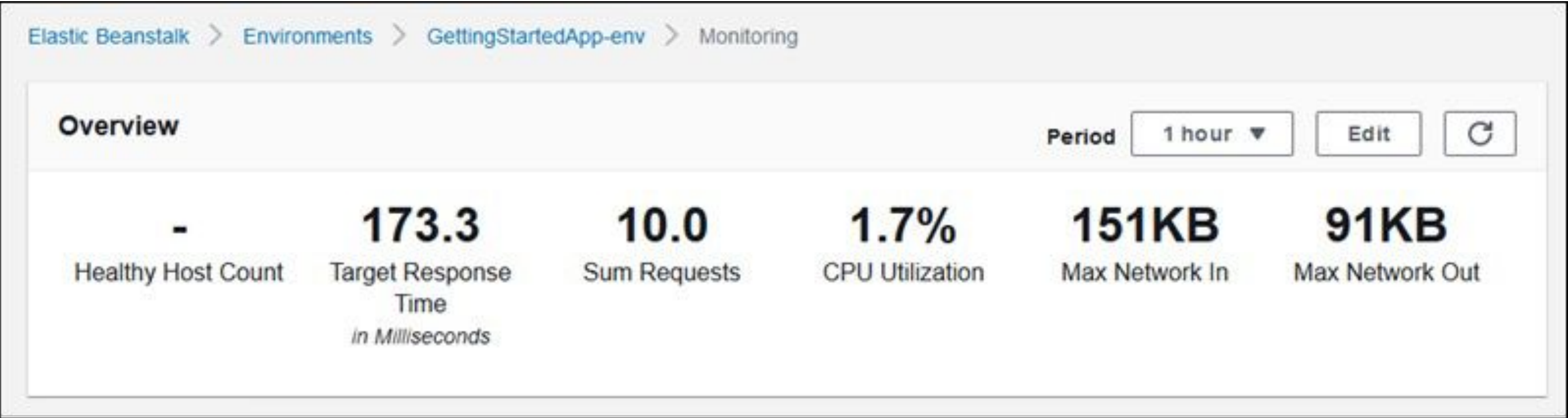
- Soporte para muchas plataformas:
  - Go
  - Java SE
  - Java con Tomcat
  - .NET en Windows Server con IIS
  - Node.js
  - PHP
  - Python
  - Ruby
  - Packer Builder
- Docker de un solo contenedor
- Docker multi contenedor
- Ventana acoplable preconfigurada
- Si no es compatible, puede configurar una plataforma personalizada (avanzado)

# DevOps Tools AWS

## Elastic Beanstalk – Monitoreo



- El agente de salud envía métricas a CloudWatch
- Comprueba el estado de la aplicación, publica eventos de estado



Recent events

Show all

< 1 >

Time	Type	Details
2020-01-28 16:06:04 UTC-0800	INFO	Environment health has transitioned from Severe to Ok.
2020-01-28 16:05:04 UTC-0800	INFO	Added instance [i-03280193ba1ba4171] to your environment.
2020-01-28 16:05:04 UTC-0800	WARN	Removed instance [i-0a4a27bbf9994ba5] from your environment due to a EC2 health check failure.
2020-01-28 16:03:04 UTC-0800	WARN	Environment health has transitioned from Ok to Severe. ELB processes are not healthy on all instances. None of the instances are sending data. ELB health is failing or not available for all instances.
2020-01-28 15:19:06 UTC-0800	INFO	Environment health has transitioned from Info to Ok. Application update completed 75 seconds ago and took 22 seconds.

Elastic Beanstalk > Environments > GettingStartedApp-env > Health

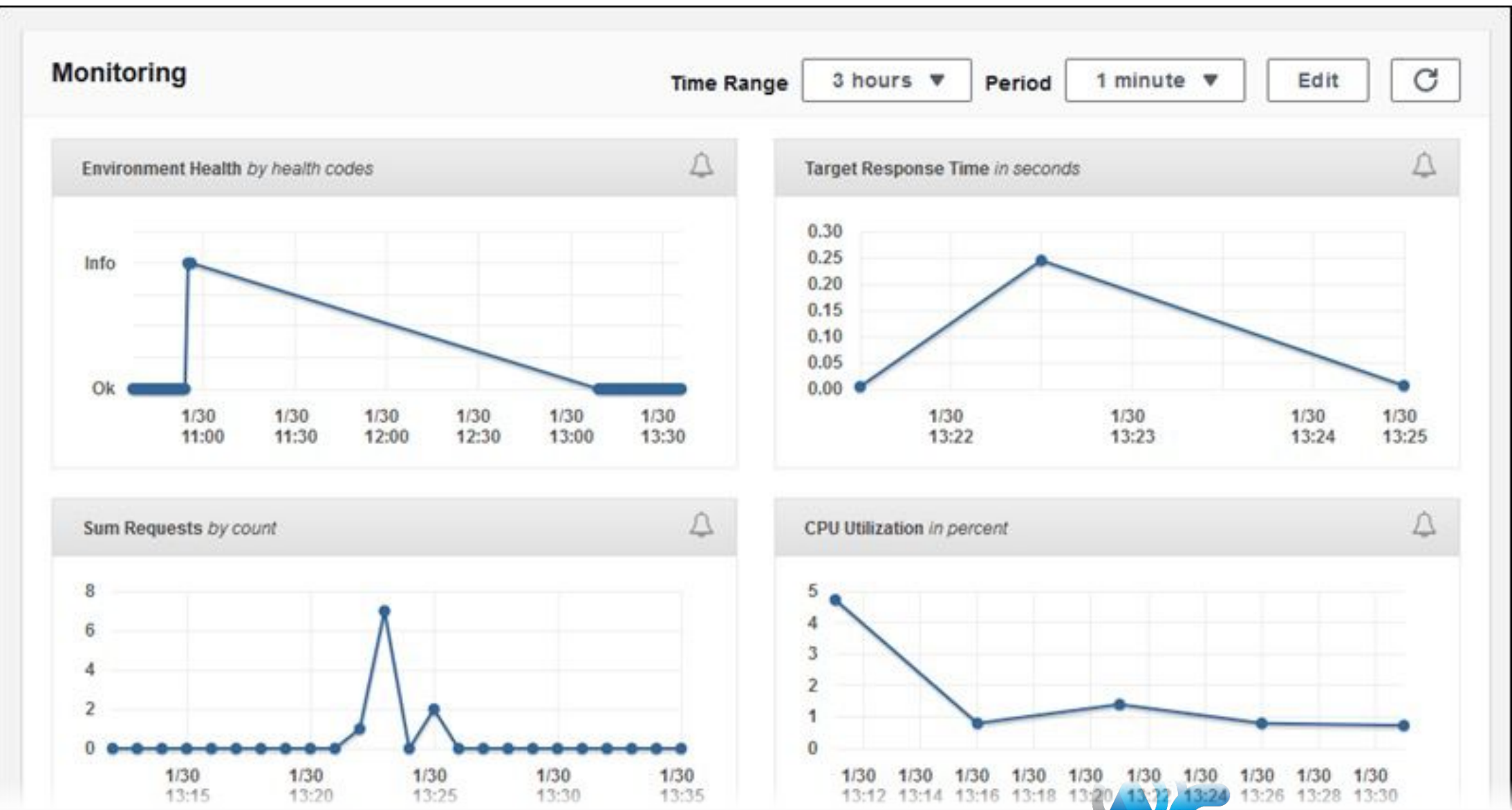
Enhanced health overview

Instances: 2 Total, 2 Ok

Learn more about enhanced health.

Filter by Instance actions

Instance ID	Status	Running	Deployment ID	Requests/sec	2xx Responses	3xx Responses	4xx Responses	5xx Responses	P99 Latency	P90 Latency	P75 Latency	P50 Latency	P10 Latency	Load1 average	Load5 average	CPU utilization User%	CPU utilization Sys%	CPU utilization Idle%	CPU utilization I/O wait%
Overall	Ok	N/A	N/A	0.4	100%	0.0%	0.0%	0.0%	0.002	0.002	0.002	0.002	0.001	N/A	N/A	N/A	N/A	N/A	N/A
i-00227707c4c4a1334	Ok	2 hours	3	0.2	2	0	0	0	0.002	0.002	0.002	0.002	0.002	0.00	0.00	0.0	0.0	99.9	0.0
i-03280193ba1ba4171	Ok	19 days	3	0.2	2	0	0	0	0.001	0.001	0.001	0.001	0.001	0.00	0.00	0.1	0.0	99.9	0.0

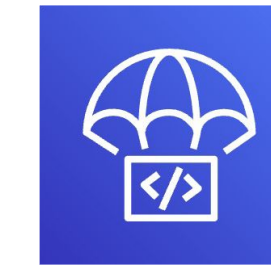




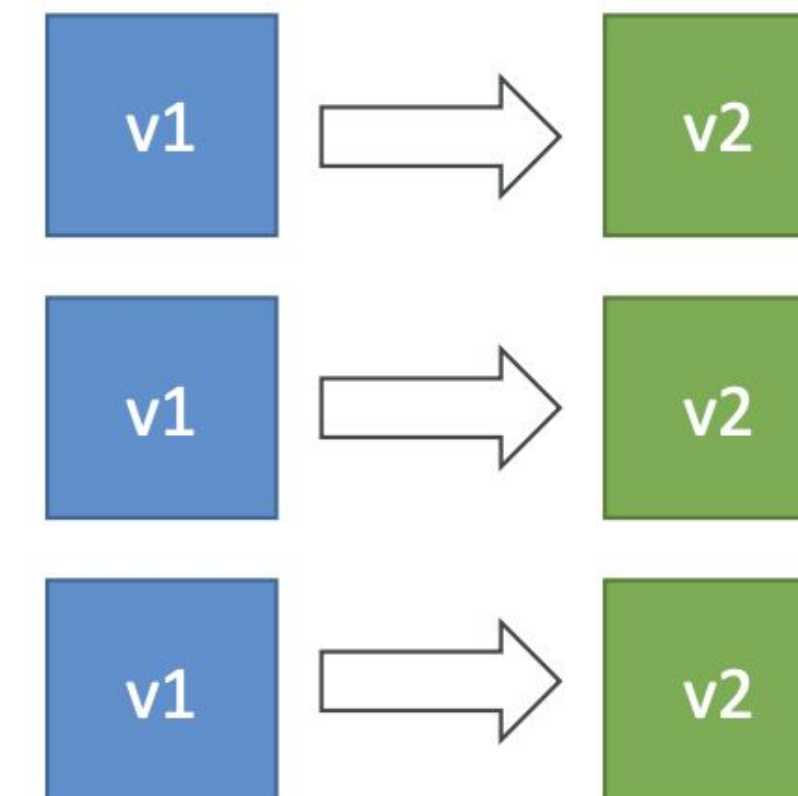
# DevOps Tools AWS

## CodeDeploy

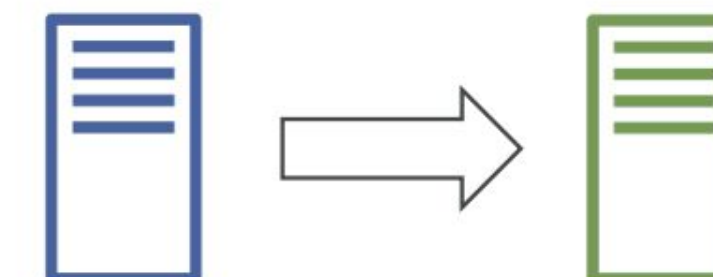
- Queremos desplegar nuestra aplicación.
- automáticamente
- Funciona con instancias EC2
- Funciona con servidores locales
- Servicio híbrido
- Los servidores/instancias deben provisionarse y configurarse con anticipación con el agente CodeDeploy.



**EC2 Instances being upgraded**



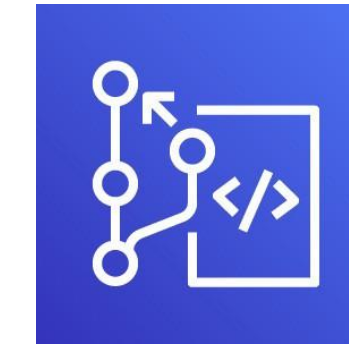
**On-premises Servers being upgraded**



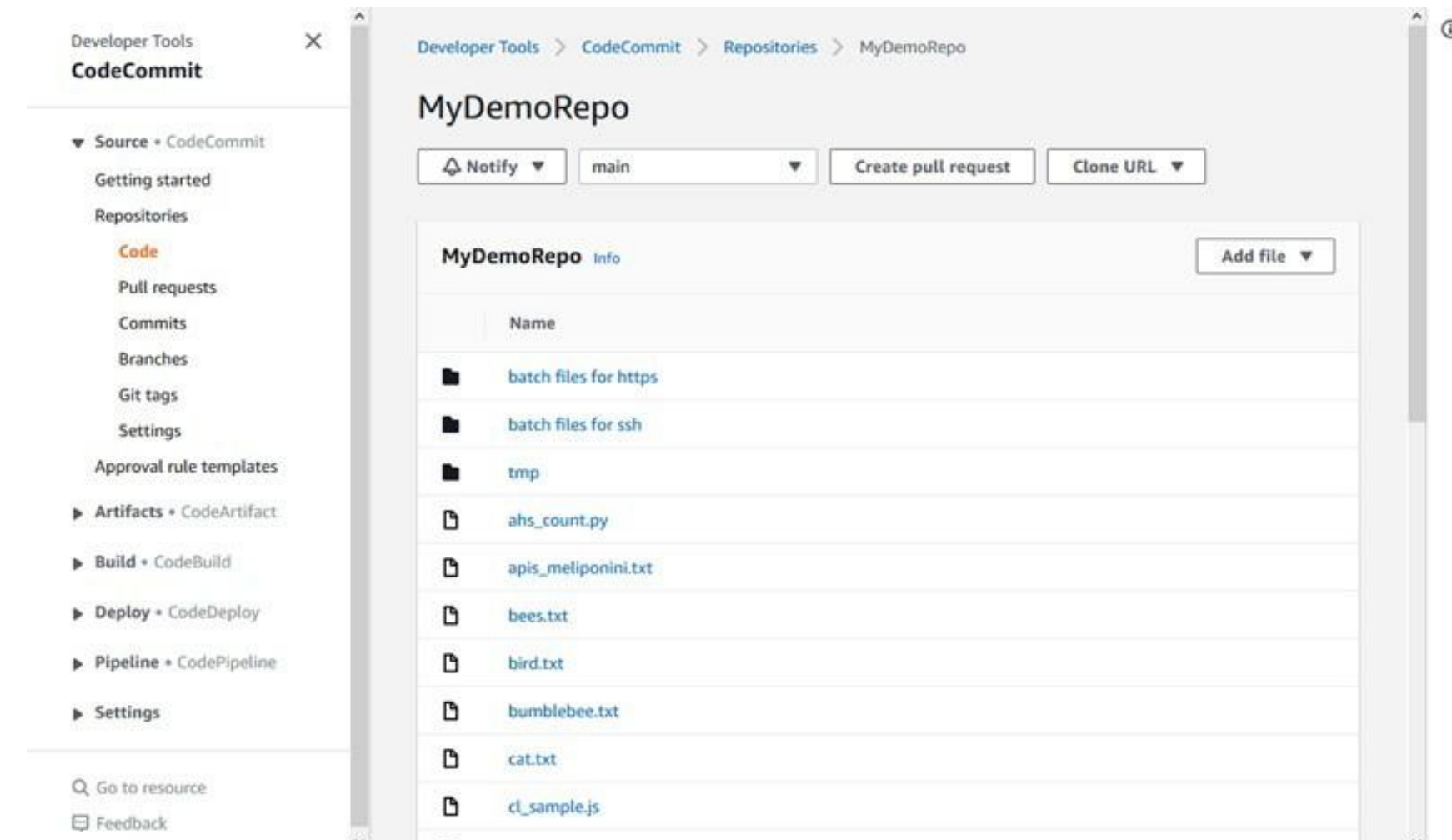


# DevOps Tools AWS

## CodeCommit

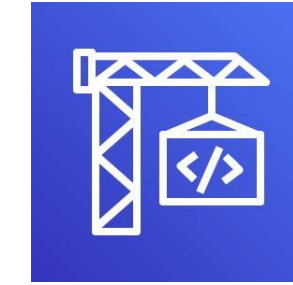


- Antes de enviar el código de la aplicación a los servidores, debe almacenarse en algún lugar.
- Los desarrolladores suelen almacenar el código en un repositorio, utilizando la tecnología Git
- Una opción conocida y similar es GitHub, el producto de la competencia de AWS es CodeCommit
- CodeCommit:
  - Servicio de control de código fuente que aloja repositorios basados en Git
  - Facilita la colaboración con otros en el código.
  - Los cambios de código se versionan automáticamente
- Beneficios:
  - Totalmente administrado
  - Escalable y de alta disponibilidad
  - Privado, protegido, integrado con AWS

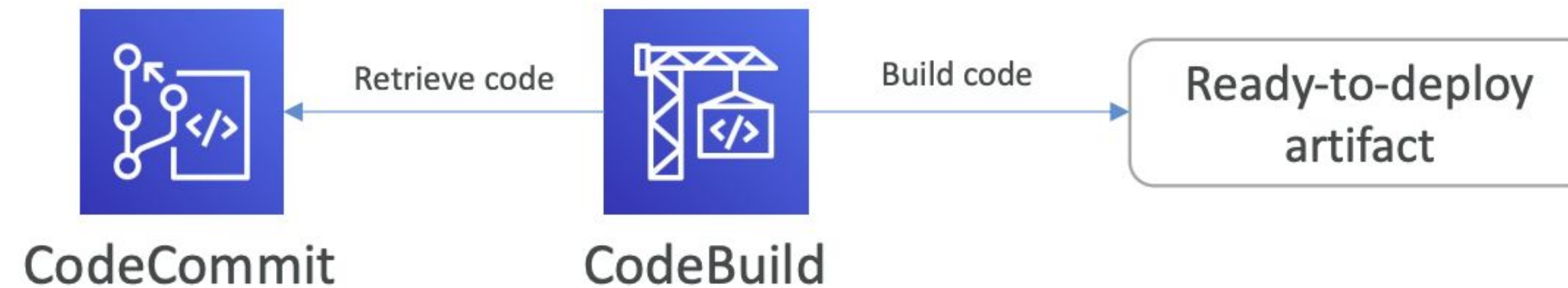


# DevOps Tools AWS

## CodeBuild



- Servicio de creación de código en la nube
- Compila el código fuente, ejecutar pruebas y produce paquetes que están listos para desplegarse (mediante CodeDeploy, por ejemplo)



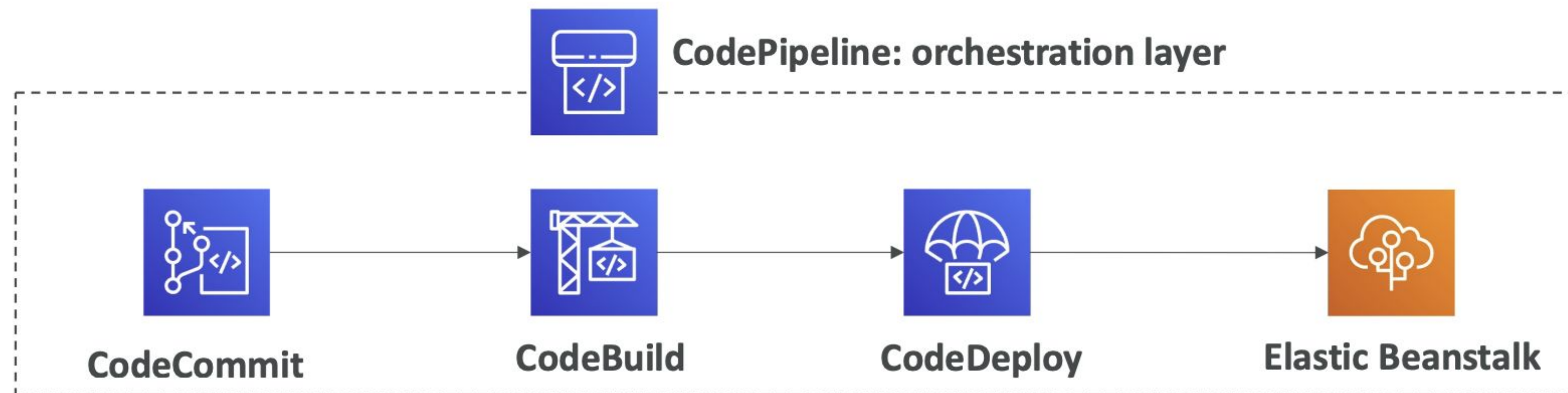
- Beneficios:
  - Totalmente administrado, sin servidor
  - Continuamente escalable y altamente disponible
  - Seguro
  - Precios de pago por uso: pague solo por el tiempo de construcción/Builds

# DevOps Tools AWS

## CodePipeline



- Organice los diferentes pasos para que el código pase automáticamente a producción
  - Código => Construir => Probar => Aprovisionar => desplegar
  - Base para CI/CD (Integración Continua y Entrega Continua)
- Beneficios:
  - Totalmente administrado, compatible con CodeCommit, CodeBuild, CodeDeploy, Elastic Beanstalk, CloudFormation, GitHub, servicios de terceros (GitHub...) y complementos personalizados...
  - Entrega rápida y actualizaciones rápidas
  - CodePipeline: capa de orquestación



# DevOps Tools AWS

## CodeArtifact



- Los paquetes de software dependen unos de otros para construirse (también llamados dependencias de código), y se crean otros nuevos
- El almacenamiento y la recuperación de estas dependencias se denomina “gestión de artefactos”
- Tradicionalmente, necesita configurar su propio sistema de gestión de artefactos
- CodeArtifact es una gestión de artefactos segura, escalable y rentable para el desarrollo de software
- Funciona con herramientas comunes de administración de dependencias como Maven, Gradle, npm, yarn, twine, pip y NuGet
- Los desarrolladores y CodeBuild pueden recuperar las dependencias directamente desde CodeArtifact

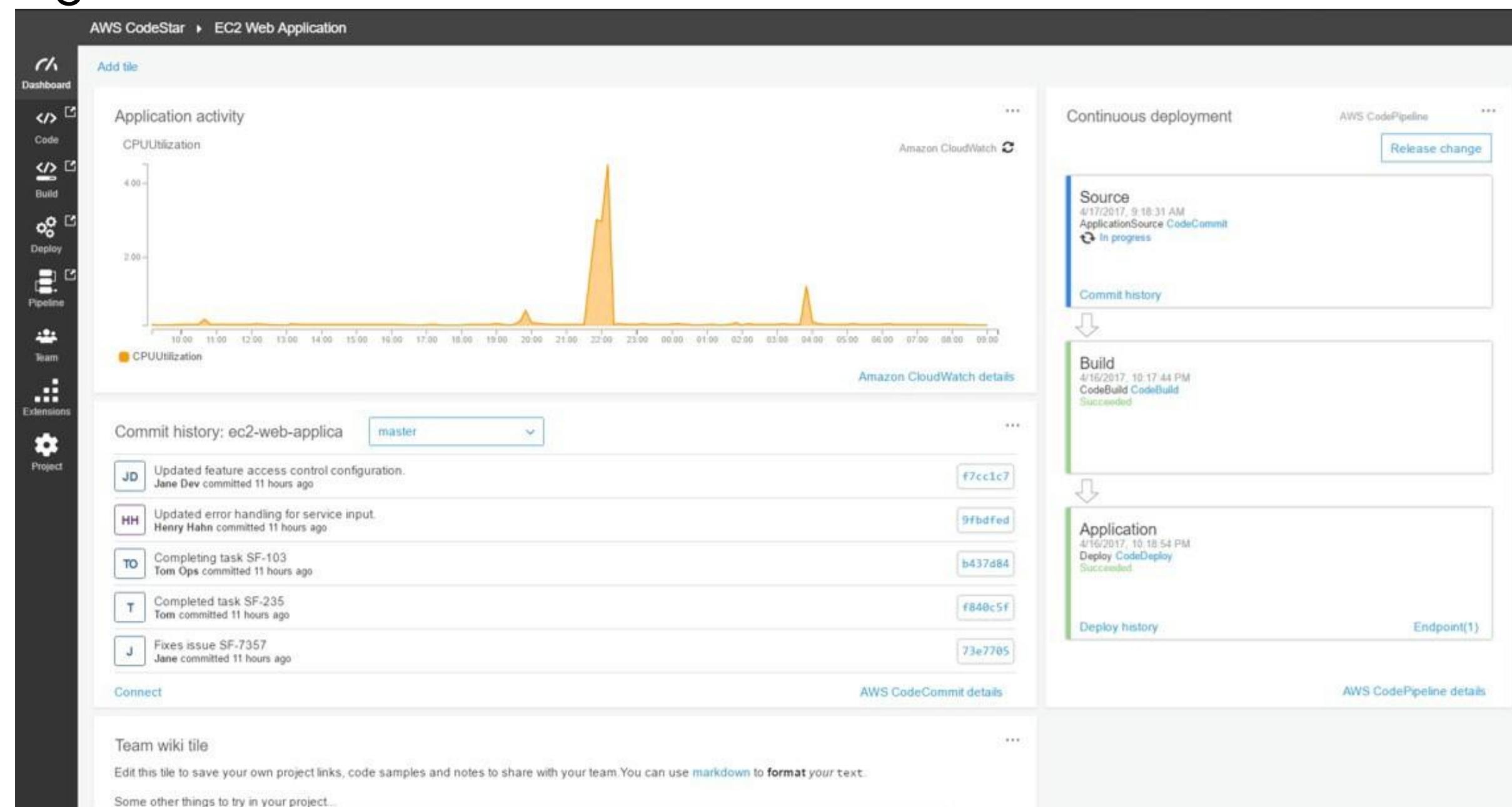


# DevOps Tools AWS

## CodeStar



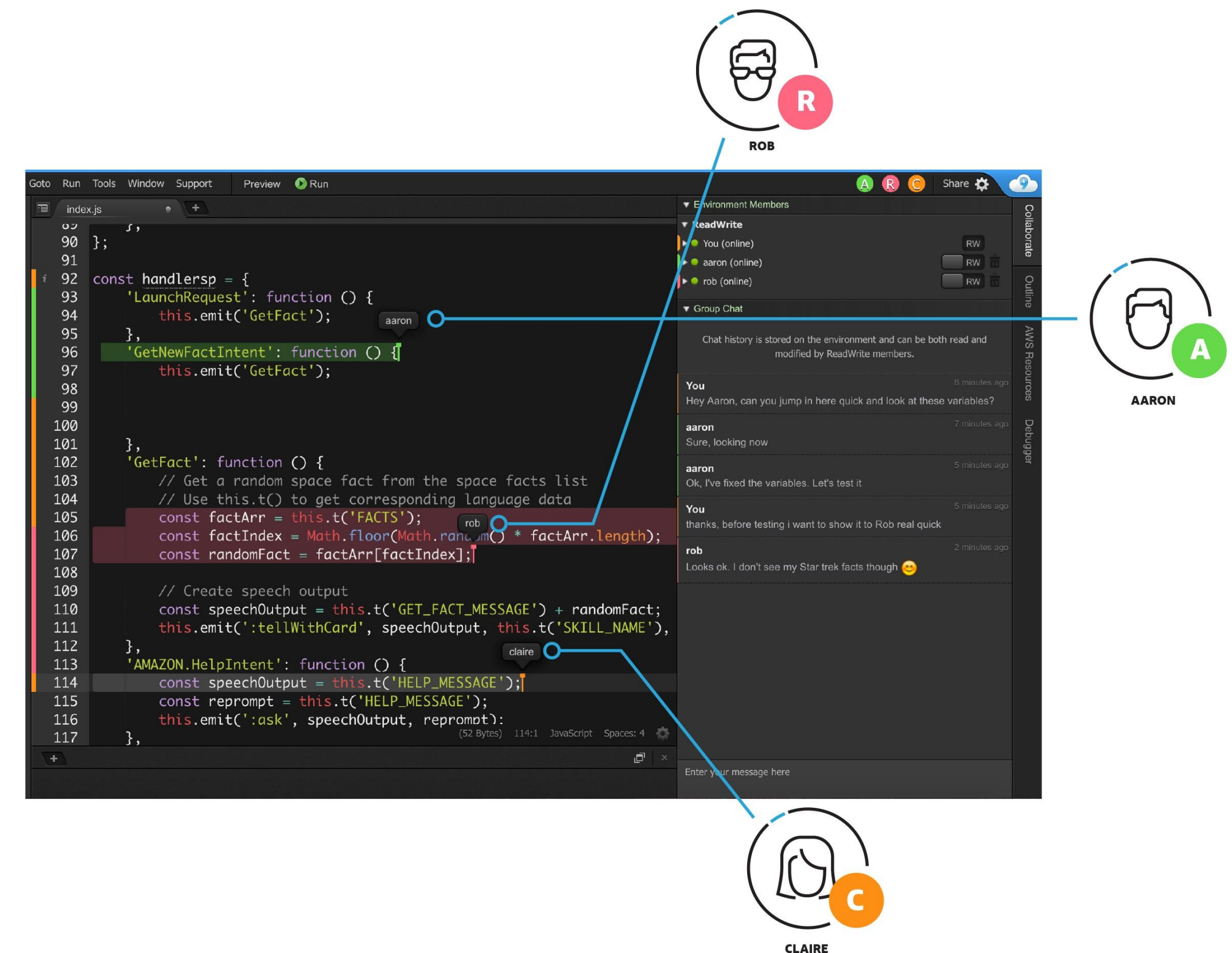
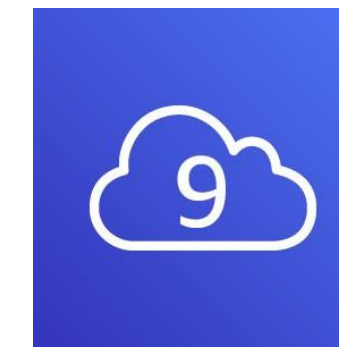
- Interfaz de usuario unificada para administrar fácilmente las actividades de desarrollo de software en un solo lugar
- “Forma rápida” de empezar a configurar correctamente CodeCommit, CodePipeline, CodeBuild, CodeDeploy, Elastic Beanstalk, EC2, etc...
- Puede editar el código "en la nube" usando AWS Cloud9



# DevOps Tools AWS

## Cloud9

- AWS Cloud9 es un IDE (entorno de desarrollo integrado) en la nube para escribir, ejecutar y depurar código
- Los IDE "clásicos" (como IntelliJ, Visual Studio Code...) se descargan en una computadora antes de usarse.
- Se puede usar un IDE en la nube dentro de un navegador web, lo que significa que puede trabajar en sus proyectos desde su oficina, hogar o cualquier lugar con Internet sin necesidad de configuración
- AWS Cloud9 también permite la colaboración de código en tiempo real (programación en pares)



# DevOps Tools AWS

## System Manager SSM



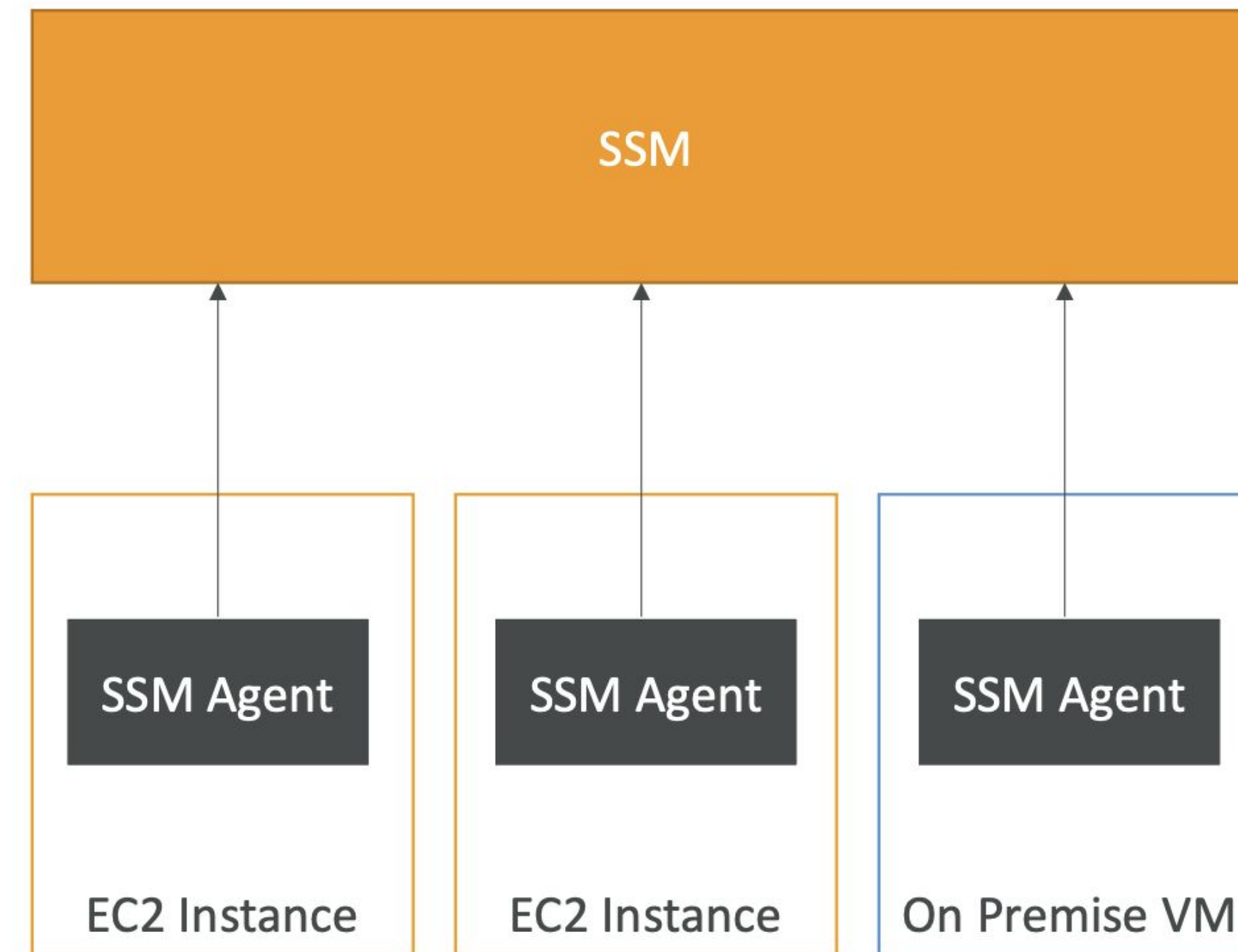
- Lo ayuda a administrar sus sistemas EC2 y On-Premises a escala
- Es otro servicio híbrido de AWS
- Sirve para obtener información operativa sobre el estado de la infraestructura
- Conjunto de más de 10 productos
- Las características más importantes son:
  - Automatización de parches
  - Ejecución de comandos en todos los servidores
  - Almacena la configuración de parámetros con SSM Parameter Store
- Funciona tanto para el sistema operativo Windows como para Linux.



# DevOps Tools AWS

## System Manager SSM – Como funciona?

- Necesitamos instalar el agente SSM en los sistemas que controlamos
- Instalado por defecto en Amazon AMI de Linux y algunas AMI de Ubuntu
- Si una instancia no se puede controlar con SSM, es probablemente un problema con el agente de SSM!
- Gracias al agente de SSM, puede ejecutar comandos, parchear y configurar nuestros servidores

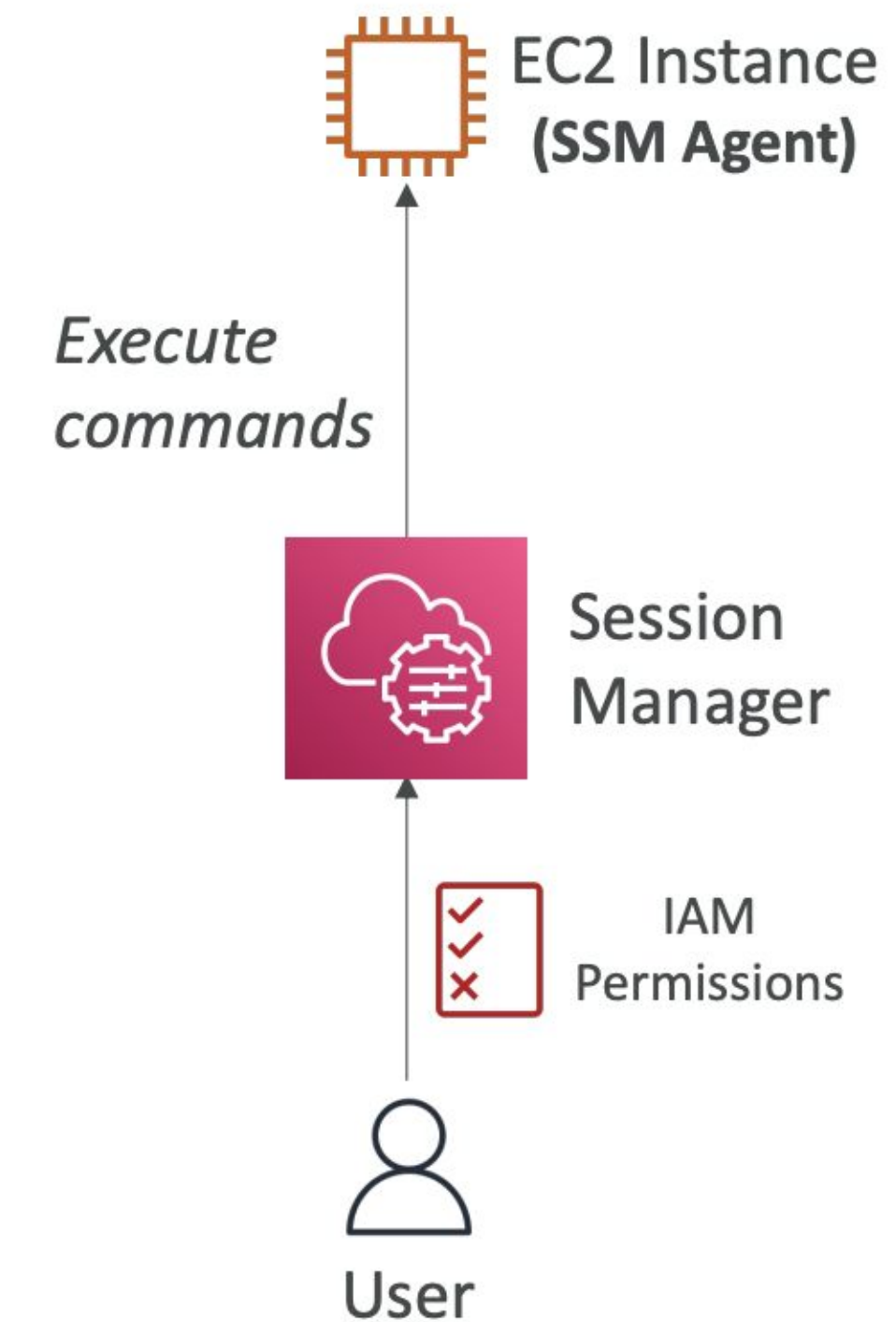




# DevOps Tools AWS

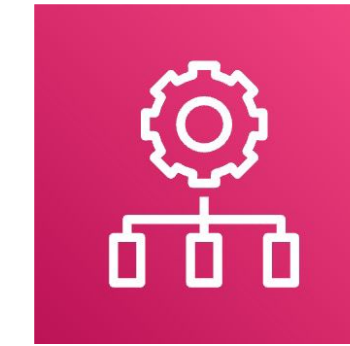
## System Manager SSM – Session Manager

- Le permite iniciar un shell seguro en su EC2 y servidores locales
- No se necesita acceso SSH, hosts bastión ni claves SSH
- No se necesita el puerto 22 (mejor seguridad)
- Compatible con Linux, macOS y Windows
- Envía datos de registro de sesión a S3 o CloudWatch Logs



# DevOps Tools AWS

## OpsWorks

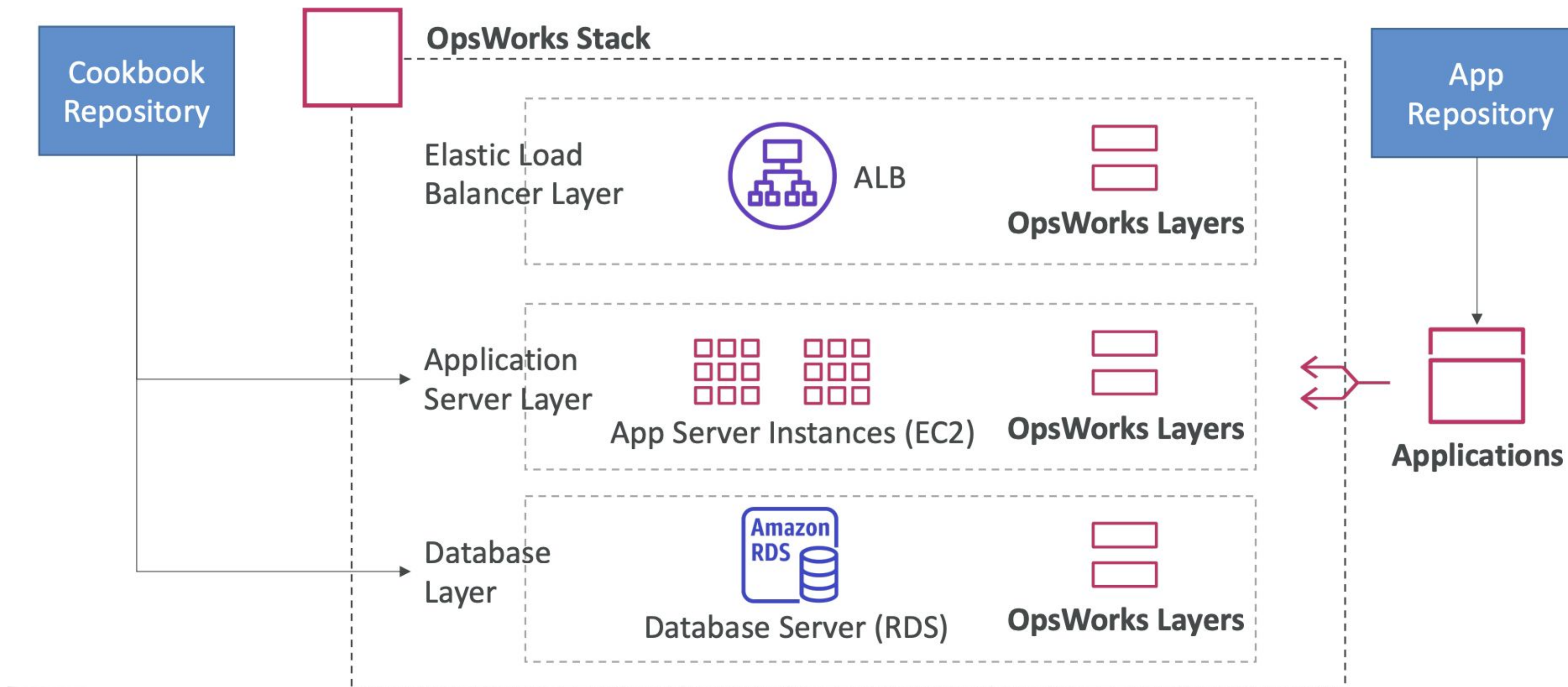


- Chef & Puppet lo ayuda a realizar la configuración del servidor automáticamente o acciones repetitivas
  - Funcionan muy bien con EC2 y máquinas virtuales locales
  - AWS OpsWorks = Chef administrado y Puppet
  - Es una alternativa a AWS SSM
  - Solo aprovisionar recursos estándar de AWS:
    - Instancias EC2, Bases de datos, Load Balancers, Volúmenes EBS...
- 
- Se necesita Chef o Puppet => AWS OpsWorks



# DevOps Tools AWS

## OpsWorks - Arquitectura



# DevOps Tools AWS

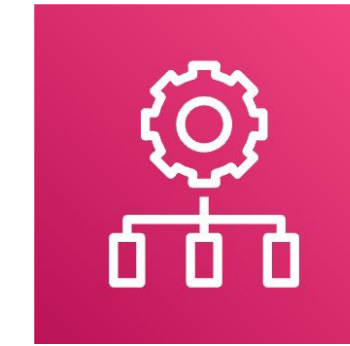
## Resumen

- CloudFormation: (solo AWS)
  - Infraestructura como código, funciona con casi todos los recursos de AWS
  - Repetir en regiones y cuentas
- Beanstalk: (solo AWS)
  - Plataforma como servicio (PaaS), limitada a ciertos lenguajes de programación o Docker
  - Implemente el código de forma coherente con una arquitectura conocida: por ejemplo, ALB + EC2 + RDS
- CodeDeploy (híbrido): implemente y actualice cualquier aplicación en servidores
- Administrador de sistemas (híbrido): aplique parches, configure y ejecute comandos a escala
- OpsWorks (híbrido): Chef y Puppet administrados en AWS



# DevOps Tools AWS

## Resumen



- CodeCommit: almacene el código en un repositorio privado de git (versión controlada)
- CodeBuild: cree y pruebe código en AWS
- CodeDeploy: desplegar código en servidores
- CodePipeline: orquestación de pipelines (desde el código hasta la compilación y la implementación)
- CodeArtifact: almacenar paquetes de software/dependencias en AWS
- CodeStar: vista unificada para permitir a los desarrolladores hacer CI/CD y código
- Cloud9: Cloud IDE (Entorno de desarrollo integrado) con colaboración
- AWS CDK: defina su infraestructura en la nube utilizando un lenguaje de programación

Ya Volvemos!  
**Un pequeño descanso.**



## Cloud Computing - AWS

# Contacto

---

[achacon@consultec-ti.com](mailto:achacon@consultec-ti.com)

 [info@consultec-ti.com](mailto:info@consultec-ti.com)

 [@consulteclatam](https://www.instagram.com/consulteclatam)

 [@consultec-ti](https://www.linkedin.com/company/consultec-ti)

 [consultec-ti.com](https://consultec-ti.com)



# Gracias

¡Nos vemos pronto!