CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

Implementación de proyecto para optimizar la gestión de enrutamiento de contenedores con Kubernetes

Project Implementation to optimize container routing management with Kubernetes

Christian Artieda [1]; Luis Aguas [2]

[1,2] Universidad Tecnológica Israel–Carrera de Sistemas de Información, 170516, Quito, Ecuador

Fecha de recepción: marzo 2022 **Fecha de aprobación:** mayo 2022

RESUMEN

Kubernetes es un sistema de orquestación de código abierto para automatizar la gestión, colocación, escalado y enrutamiento de contenedores que se ha popularizado entre los desarrolladores y los equipos de operaciones de TI en los últimos años. Primero fue desarrollado por Google y contribuyó a Open Source en 2014, y ahora es mantenido por la Cloud Native Computing Foundation. Existe una comunidad y un ecosistema de Kubernetes activos que se desarrollan alrededor de Kubernetes con miles de contribuyentes y docenas de socios certificados.

Palabras clave: Kubernetes, Source, Linux, OpenShift.

ABSTRACT

Kubernetes is an open source orchestration system to automate container management, placement, scaling, and routing that has become popular with developers and IT operations teams in recent years. It was first developed by Google and contributed to Open Source in 2014, and is now maintained by the Cloud Native Computing Foundation. There is an active Kubernetes community and ecosystem that thrives around Kubernetes with thousands of contributors and dozens of certified partners.

Key Words: Kubernetes, Source, Linux, OpenShift

[1] Estudiante de Ingeniería en Sistemas, kristian.ldu@gmail.com
[2] Magíster en Redes de Comunicaciones, aguaszoft@outlook.es

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

1. INTRODUCCIÓN

Kubernetes es una aplicación para máquinas MacOS y Windows para construir y compartir aplicaciones en contenedores y microservicios [1].

Docker Desktop ofrece la velocidad, las opciones y la seguridad que necesita para diseñar y entregar estas aplicaciones en contenedores en su escritorio. Docker Desktop incluye la aplicación Docker, herramientas de desarrollo, Kubernetes y sincronización de versiones para motores Docker de producción.

Además, permite aprovechar imágenes y plantillas certificadas y su elección de idiomas y herramientas. Los flujos de trabajo de desarrollo aprovechan Docker Hub para extender su entorno de desarrollo a un repositorio seguro para una rápida construcción automática, integración continua y colaboración segura.

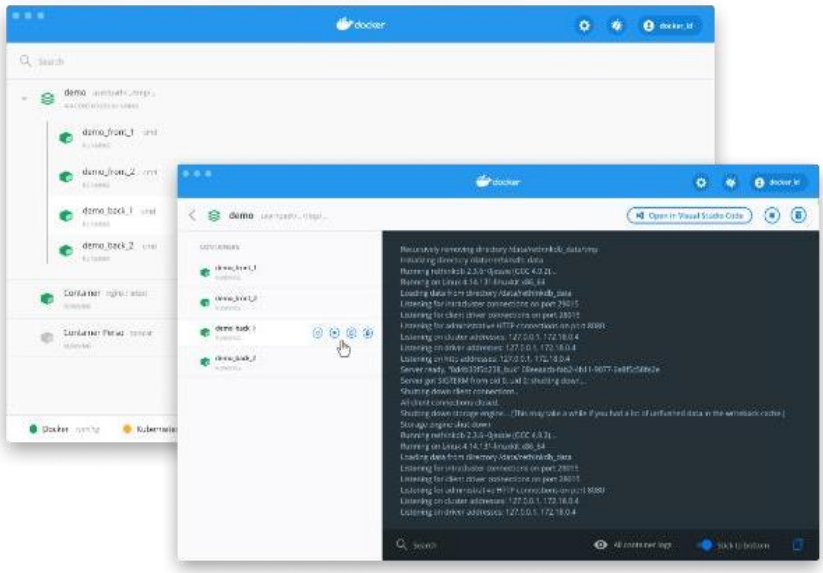

Figura 1

2. DESARROLLO

2.1 Importancia de implementar Kubernetes

Las aplicaciones de producción real abarcan varios contenedores. Esos contenedores deben implementarse en varios hosts de servidores. La seguridad de los contenedores tiene varias capas y puede ser complicada. Aquí es donde Kubernetes puede ayudarlo. Kubernetes le ofrece la capacidad de organización y gestión necesaria para implementar contenedores a escala para estas cargas de trabajo.

El sistema de organización de Kubernetes le permite diseñar servicios de aplicaciones que abarcan varios contenedores, programar esos contenedores en un clúster, ampliarlos y gestionar su estado a lo largo del tiempo. Con Kubernetes, puede adoptar medidas concretas para lograr una mejor seguridad de TI [1].

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

Kubernetes también debe integrarse a las conexiones en red, el almacenamiento, la seguridad, la telemetría y otros servicios para proporcionar una infraestructura de contenedores integral.

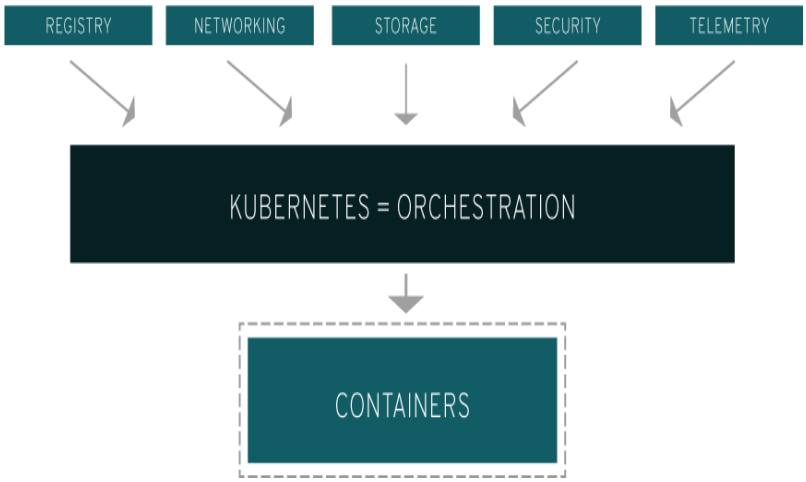

Figura 2

2.2 Cómo funciona y para qué sirve Kubernetes

La principal ventaja de usar Kubernetes en su entorno, especialmente si se encuentra optimizando el desarrollo de las aplicaciones para la nube, es que le ofrece la plataforma para programar y ejecutar contenedores en clústeres de máquinas virtuales o físicas [2]. A grandes rasgos, le permite implementar una infraestructura basada en contenedores en los entornos de producción, y depender completamente de ella. Y dado que Kubernetes abarca todo lo referido a la automatización de tareas operativas, puede hacer muchas de las cosas que también otras plataformas de aplicaciones o sistemas de gestión le permiten hacer, pero para sus contenedores.

Con Kubernetes se puede:

- Orquestar contenedores en múltiples hosts.
- Hacer un mejor uso del hardware para maximizar los recursos necesarios para ejecutar sus aplicaciones empresariales.
- Controlar y automatizar las implementaciones y actualizaciones de las aplicaciones.
- Montar y añadir almacenamiento para ejecutar aplicaciones con estado.
- Escalar las aplicaciones en contenedores y sus recursos sobre la marcha.
- Administrar servicios de forma declarativa, que garanticen que las aplicaciones implementadas siempre se ejecuten del modo que las implementó.
- Comprobaciones de estado y autorregeneración de sus aplicaciones con ubicación, reinicio, replicación y escalamiento automáticos.

2.3 Uso de Kubernetes en la producción

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

Kubernetes es open source, de hecho, no hay una estructura de soporte formalizada en torno a esa tecnología, o por lo menos no la que le gustaría para su empresa. Si tuvo problemas para implementar Kubernetes durante su ejecución en la producción, no estará de lo más contento, y probablemente, sus clientes tampoco [3].

Allí es donde Red Hat OpenShift entra en acción. OpenShift es Kubernetes para la empresa y mucho más. OpenShift comprende todas las piezas de tecnología adicionales que convierten a Kubernetes en una herramienta sólida y viable para las empresas, lo que incluye lo siguiente: registro, conexiones en red, telemetría, seguridad, automatización y servicios, entre otras características. Con OpenShift, sus desarrolladores pueden crear nuevas aplicaciones en contenedores, alojarlas e implementarlas en la nube, con la escalabilidad, el control y la orquestación que pueden transformar una buena idea en nuevos negocios de forma fácil y rápida.

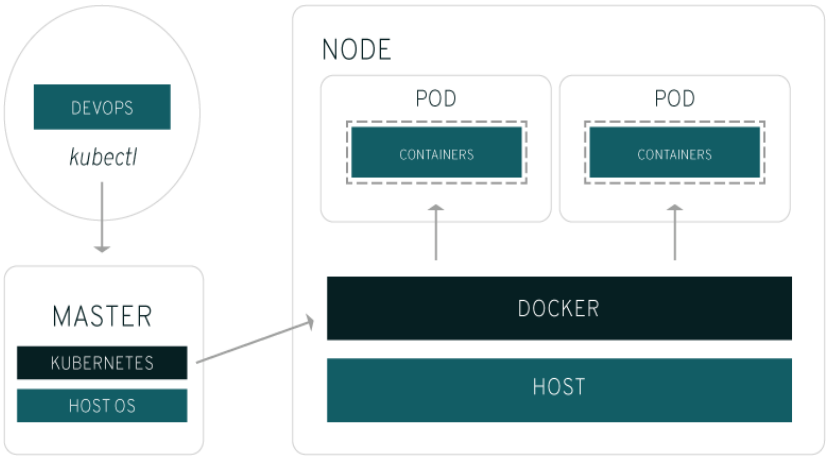

Figura 3.

2.4 Arquitectura

Kubernetes sigue una arquitectura maestro-esclavo. Los componentes de Kubernetes pueden ser divididos en aquellos que administran un nodo individual y aquellos que son partes de un panel de control.

2.4.1 Etcd

Es un almacén de datos persistente, liviano, distribuido de clave-valor desarrollado por CoreOS que almacena de manera confiable los datos de configuración del clúster, representando el estado general del cluster en un punto del tiempo dado. Otros componentes escuchan por cambios en este almacén para avanzar al estado deseado.

2.4.2 Servidor de API

El servidor API es un componente central y sirve a la API de Kubernetes utilizando JSON sobre HTTP, que proveen la interfaz interna y externa de Kubernetes1321. El servidor API procesa y valida las peticiones REST y actualiza el estado de los objetos

API en etcd, así permitiendo a los clientes configurar flujos de trabajos y contenedores a través de los nodos esclavos.

2.4.3 Planificador

El planificador es el componente enchufable que selecciona sobre qué nodo deberá correr un pod sin planificar (la unidad básica de manejo del planificador) basado en la disponibilidad de recursos. El planificador lleva la cuenta de la utilización de recursos en cada nodo para asegurar que un flujo de trabajo no es planificado en exceso de la disponibilidad de los recursos. Para este propósito, el planificador debe conocer los requerimientos de recursos, la disponibilidad de recursos y una variedad de restricciones y políticas directivas como quality-of-service (QoS), requerimiento de afinidad, localización de datos entre otros. En esencia, el rol del planificador es el de emparejar la oferta de un recurso con la demanda de un flujo de trabajos.

2.5 Administrador del controlador

El administrador de controlador es el proceso sobre el cual el núcleo de los controladores Kubernetes como DaemonSet y Replication se ejecuta. Los controladores se comunican con el servidor API para crear, actualizar y eliminar recursos que ellos manejan (pods, servicios, etc.)

2.5.1 Nodo Kubernetes

El nodo, también conocido como esclavo o worker, es la máquina física (o virtual) donde los contenedores (flujos de trabajos) son desplegados. Cada nodo en el clúster debe ejecutar la rutina de tiempo de ejecución (como Docker), así como también los componentes mencionados más abajo, para comunicarse con el maestro para la configuración en red de estos contenedores.

2.5.2 Kubelet

Es responsable por el estado de ejecución de cada nodo (es decir, asegurarse que todos los contenedores en el nodo se encuentran saludables). Se encarga del inicio, la detención y el mantenimiento de los contenedores de aplicaciones (organizados como pods) como es indicado por el panel de control.

Monitorea el estado de un pod y, si no se encuentra en el estado deseado, el pod será desplegado nuevamente al mismo nodo. El estado del nodo es comunicado al maestro cada poco segundo mediante una señal periódica ("heartbeat"). Una vez que el nodo detecta la falla de un nodo, el Replication Controller observa este cambio de estado y lanza pods en otros nodos sanos.

2.5.3 Kube-Proxy

Kube-Proxy es la implementación de un proxy de red y balanceador de carga soportando la abstracción del servicio junto con otras operaciones de red13. Es

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

responsable del enrutamiento del tráfico hacia el contenedor correcto basado en la dirección IP y el número de puerto indicados en el pedido.

2.5.4 CAdvisor

Es un agente que monitorea y recoge métricas de utilización de recursos y rendimiento como CPU, memoria, uso de archivos y red de los contenedores en cada nodo.

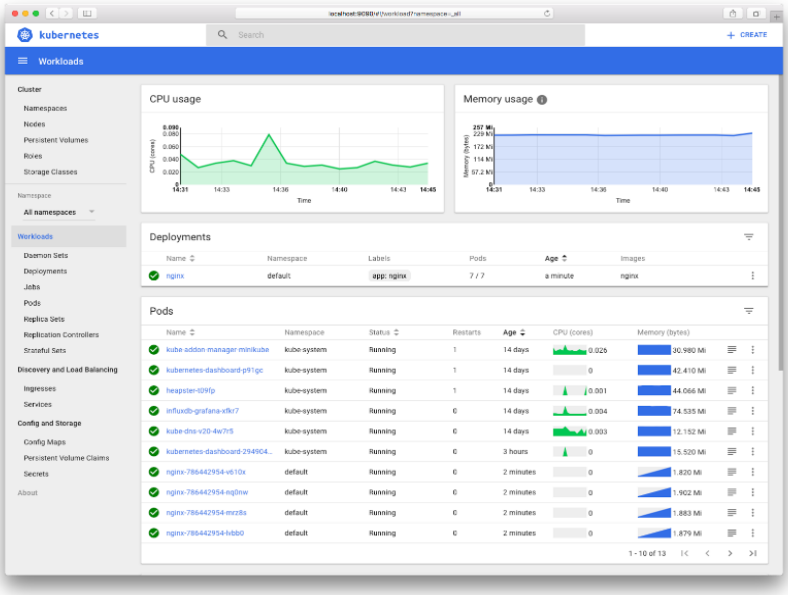

Figura 4.

2.6 Dónde utilizar Kubernetes

Las opciones para utilizar Kubernetes apenas tienen restricción, casi cualquier opción de uso es posible gracias a todas las posibilidades de instalación que ofrece y porque muchas soluciones lo están integrando en sus arquitecturas. Así, disponemos de un abanico amplio para utilizar K8S.

1. Bare Metal: podemos desplegar nuestro cluster directamente sobre máquinas físicas utilizando múltiples sistemas operativos: Fedora, CentOS, Ubuntu, etc.
2. Virtualización On Premise: si queremos montar nuestro cluster on premise, pero con máquinas virtuales, las posibilidades crecen. Podemos utilizar Vagrant, CloudStack, Vmware, OpenStack, CoreOS, oVirt, Fedora, etc.
3. Soluciones Cloud: si queremos disponer de todas las ventajas de Kubernetes, sin encargarnos de gestionar todo lo que hay por debajo, tenemos todas estas alternativas en la nube:
   - Google Container Engine: servicio gestionado y ofrecido por Google, que se encarga de manejar por nosotros las instancias de Compute Engine que tiene por debajo. Se ocupa también de la monitorización, del logging, de la salud de las instancias y de actualizar Kubernetes a la última versión disponible.

- OpenShift: el líder de los PaaS integra Kubernetes y, al utilizarlo en sus diferentes opciones (enterprise, online, etc.), estaremos haciendo uso de clusters gestionados de K8S.
- CoreOS Tectonic: es el producto mediante el que CoreOS proporciona Kubernetes. Facilita la portabilidad, entre varios proveedores, de cloud pública y privada. Su instalación, actualización y mantenimiento requieren de menos trabajo de operaciones. Incluye Prometheus para la monitorización y gestión de alertas.
- CoreOS: van a sustituir su sistema fleet por Kubernetes.
- Kops: sirve para crear y gestionar clusters de Kubernetes (si se requiere, en producción y con alta disponibilidad) desde línea de comandos. Hasta el momento, ha sido la forma oficiosa para instalar Kubernetes en AWS y tienen en sus planes abarcar también Google Compute Engine y VMware vSphere.
- Deis: su PaaS opensource, que ahora se llama Workflow, se basa en Kubernetes desde hace años.
- Mesosphere: parece que, tras su último anuncio, se van a inclinar mucho por el uso de Kubernetes como orquestador en lugar de Marathon.
- CloudFoundry ofrece Kubernetes en su Container Runtime.
- Otros: Azure, IBM, Kube2Go, GiantSwarm también ofrecen servicios gestionados de kubernetes.

Figura 5.

2.7 Ejemplo de flujo de trabajo de DevOps con Kubernetes

Repita, prueba y depure rápidamente diferentes partes de una aplicación juntas en el mismo clúster de Kubernetes.

Fusione código mediante combinación e insértelo en un repositorio de GitHub para lograr integración continua. A continuación, ejecute compilaciones y pruebas automatizadas como parte de la entrega continua.

Compruebe el origen y la integridad de las imágenes de contenedor. Las imágenes se mantienen en cuarentena hasta que pasan un examen.

Aprovisione clústeres de Kubernetes con herramientas como Terraform. Gráficos Helm instalados por Terraform definen el estado deseado de los recursos y las configuraciones de las aplicaciones.

Imponga directivas para gobernar las implementaciones en el clúster de Kubernetes.

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

La canalización de versión ejecuta automáticamente una estrategia de implementación predefinida con cada código.

Agregue auditoría de directivas y corrección automática a la canalización de CI/CD. Por ejemplo, solo la canalización de versión tiene permiso para crear nuevos pods en su entorno de Kubernetes.

Habilite telemetría de las aplicaciones, supervisión del estado de los contenedores y análisis de registros en tiempo real.

Solucione los problemas con conclusiones e informe de los planes para el próximo sprint.

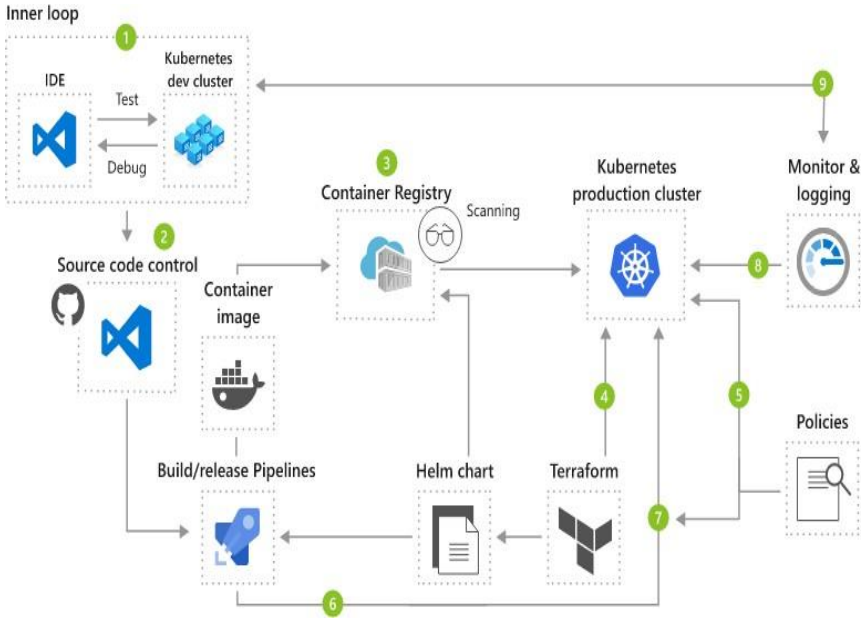

Figura 6.

2.8 Uso de Kubernetes

Kubernetes se ha convertido en la plataforma de orquestación estándar para contenedores. Todos los principales proveedores de la nube lo admiten, por lo que es la opción lógica para las organizaciones que buscan mover más aplicaciones a la nube [4].

Kubernetes proporciona un marco común para ejecutar sistemas distribuidos para que los equipos de desarrollo tengan una infraestructura constante e inmutable desde el desarrollo hasta la producción para cada proyecto. Kubernetes puede gestionar los requisitos de escala, disponibilidad, conmutación por error, patrones de implementación y más.

2.8.1 Capacidades de Kubernetes

- Servicio y definición de procesos
- Servicio de descubrimiento y equilibrio de carga
- Orquestación de almacenamiento

CREANDO INGENIOS
ISSN: 3028-8924
Correo: editor.revista@tecnologicoismac.edu.ec
URL: https://ismaconline.net/investigacion/index.php/CreaIngenio_2021/index
Volumen 2, Número 1 / Enero – Junio 2022 pp. 39-48

- Gestión de recursos a nivel de contenedor
- Despliegue automatizado y reversión
- Gestión de salud de contenedores
- Secretos y gestión de configuraciones

2.8.2 Ventajas de Kubernetes

Kubernetes tiene muchas capacidades potentes y avanzadas. Para los equipos que tienen las habilidades y el conocimiento para aprovecharlo al máximo, Kubernetes ofrece:

Disponibilidad. La agrupación de Kubernetes tiene incorporada una tolerancia a fallas muy alta, lo que permite operaciones a gran escala.
Autoescalado. Kubernetes puede ampliar y reducir según el tráfico y la carga del servidor automáticamente.

Amplio ecosistema. Kubernetes tiene un ecosistema sólido en torno a la Interfaz de red de contenedores (CNI) y la Interfaz de almacenamiento de contenedores (CSI) y las herramientas incorporadas de registro y monitoreo.

Sin embargo, la complejidad de Kubernetes es abrumadora para muchas personas que saltan por primera vez. Los primeros en adoptar Kubernetes han sido sofisticados grupos tribales de desarrolladores de organizaciones de mayor escala con una cultura de bricolaje y fuertes equipos de desarrolladores independientes con las habilidades para "rodar sus propios" Kubernetes.

A medida que la corriente principal comienza a considerar la adopción de Kubernetes internamente, este enfoque es a menudo lo que se hace referencia en la comunidad en general en la actualidad. Pero este enfoque puede no ser adecuado para todas las organizaciones.

Si bien Kubernetes tiene capacidades avanzadas, todo ese poder tiene un precio; saltar a la cabina de un avión de última generación pone mucho poder debajo de ti, pero no es obvio cómo volar realmente.

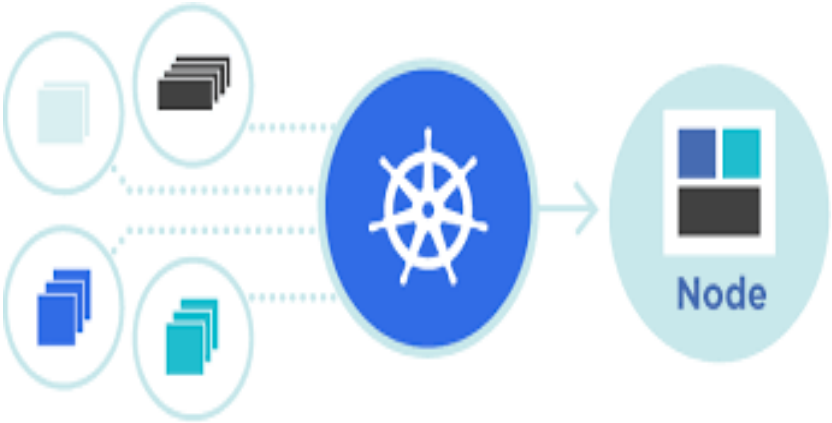

Figura 7.

3. CONCLUSIONES

El impacto que está teniendo Kubernetes en la orquestación y gestión de entornos basados en contenedores, que se ha convertido en la herramienta más utilizada por varias soluciones, e incluso Mesosphere lo ha implementado sustituyendo a su orquestador original Marathon.

Actualmente se encuentra en un momento de gran crecimiento gracias a la comunidad apoyada por Google, Red Hat, Mesosphere, CoreOS, AWS, etc. por lo que las mejoras son continuas.

Los contenedores han supuesto una revolución en la forma de desarrollar y desplegar aplicaciones, pero esto no sería posible sin herramientas como Kubernetes. Por lo tanto, esperamos que se termine de consolidar en el mercado y no pare de crecer ya que tiene un gran recorrido por delante.

REFERENCIAS

1. López, F. (2019, 16 enero). Kubernetes. Servicios y soluciones IT Conasa. https://www.conasa.es/blog/kubernetes/
2. Por qué todos apuestan por Kubernetes. (2020, 12 mayo). Paradigma. https://www.paradigmadigital.com/techbiz/por-que-todos-apuestan-por-kubernetes/
3. C. Multi-cloud Kubernetes on. (2020). Ubuntu. https://ubuntu.com/kubernetes
4. O'Brien ¿Qué es Kubernetes? (2020). Microsoft Azure. https://azure.microsoft.com/es-es/topic/what-is-kubernetes/#azure-kubernetes-service
   Y. Chen, L. Ge, B. https://www.docker.com/products/kubernet