



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Revisión Sistemática de la Literatura Sobre Tecnologías Docker

Yuliana Franco Noreña

Universidad Nacional de Colombia
Facultad de Minas, Área Curricular de Sistemas e Informática
Medellín, Colombia
2022

Revisión Sistemática de la Literatura Sobre Tecnologías Docker

Yuliana Franco Noreña

Trabajo Final presentado como requisito parcial para optar al título de:
Magister en Ingeniería Analítica

Director:

Juan David Velásquez Henao, PhD

Línea de Investigación:

Analítica

Grupo de Investigación:

Big Data & Data Analytics

Universidad Nacional de Colombia

Facultad de Minas, Área Curricular de Sistemas e Informática

Medellín, Colombia

2022

Declaración de obra original

Yo declaro lo siguiente:

He leído el Acuerdo 035 de 2003 del Consejo Académico de la Universidad Nacional. «Reglamento sobre propiedad intelectual» y la Normatividad Nacional relacionada al respeto de los derechos de autor. Esta disertación representa mi trabajo original, excepto donde he reconocido las ideas, las palabras, o materiales de otros autores.

Cuando se han presentado ideas o palabras de otros autores en esta disertación, he realizado su respectivo reconocimiento aplicando correctamente los esquemas de citas y referencias bibliográficas en el estilo requerido.

He obtenido el permiso del autor o editor para incluir cualquier material con derechos de autor (por ejemplo, tablas, figuras, instrumentos de encuesta o grandes porciones de texto).

Por último, he sometido esta disertación a la herramienta de integridad académica, definida por la universidad.

YULIANA FRANCO NOREÑA

Nombre

27/01/2022

Fecha

Agradecimientos

Deseo expresar mi agradecimiento al director de esta tesis de maestría, profesor Juan David Velásquez Henao, por la dedicación y orientación que ha brindado a este trabajo, desde el primer momento impulsando y motivando mi confianza académica y profesional; por supuesto, gracias a mi familia y amigos por su apoyo incondicional y entender el esfuerzo de tiempo y momentos dedicados para sacar adelante este sueño profesional.

Resumen

Los contenedores Docker han demostrado ser una tecnología eficiente para el despliegue de aplicaciones; ofreciendo atención a desafíos de seguridad, portabilidad y escalabilidad; además, es una herramienta vital para instalar infraestructuras de microservicios. El propósito de este trabajo es desarrollar la revisión sistemática de la literatura acerca de esta tecnología, la cual adopta cada vez más, una mayor participación en el mundo del desarrollo de software; para ello, se emplean diferentes técnicas de análisis y minería de texto para identificar los principales temas que se consideran una tendencia en este campo de investigación. El estudio se lleva a cabo sobre las palabras claves de los artículos, mediante las cuales se evalúan todos aquellos con mayor número de citas; pero también, se incluyen los artículos no citados; como resultado, el análisis revela como la producción de artículos para este tema se centraliza principalmente en países como Estados Unidos, China y la India; también, su dominio de investigación se dirige hacia temas de la computación de alto rendimiento, arquitecturas con microservicios, el internet de las cosas y plataformas usadas para la administración de contenedores.

Palabras clave: (docker, container, dockers-containers).

Abstract

Title: Systematic Review of the Literature on Docker Technologies

Docker containers have proven to be an efficient technology for the deployment of applications; offering attention to security, portability and scalability challenges; in addition, it is a vital tool for installing microservices infrastructures. The purpose of this paper is to develop a systematic review of the literature about this technology, which increasingly adopts a greater participation in the world of software development; for this purpose, different analysis and text mining techniques are used to identify the main topics that are considered a trend in this field of research. The study is carried out on the keywords of the articles, through which all those with the highest number of citations are evaluated; but also, uncited articles are included; as a result, the analysis reveals how the production of articles for this topic is mainly centralized in countries such as the United States, China and India; also, its research domain is directed towards topics of high performance computing, architectures with microservices, the internet of things and platforms used for container management.

Keywords: (docker, container, dockers-containers).

Contenido

	<u>Pág.</u>
Resumen	V
Lista de figuras	IX
Lista de tablas	X
1. Introducción	11
1.1 Protocolo de Investigación	13
1.2 Hipótesis	15
1.3 Objetivos	15
1.3.1 Objetivo General.....	15
1.3.2 Objetivos Específicos.....	15
1.4 Mapa del documento.....	15
2. Conceptos Básicos	16
2.1 Docker.....	16
2.2 Máquina Virtual	16
2.3 Contenedor	16
2.4 Imagen de Docker.....	17
2.5 Kubernetes.....	17
2.6 Docker Hub	17
2.7 Dockerfile	18
2.8 Docker Client (Cliente de Docker)	18
2.9 Docker Engine.....	18
2.10 Docker Daemon	18
3. Resultados Obtenidos	19
3.1 Estadísticas básicas.....	19
3.2 Principales Revistas	21
3.3 Principales Autores	22
3.4 Publicaciones por país	25
3.5 Artículos más citados	26
4. Determinación y caracterización de las áreas temáticas sobre tecnologías Docker	31
4.1 Determinación de las áreas temáticas.....	31
4.2 Caracterización de los clústeres.....	33
4.2.1 Clúster 1	33
4.2.2 Clúster 2	34
4.2.3 Clúster 3	35

4.2.4 Clúster 4	36
5. Conclusiones y recomendaciones.....	38
5.1 Conclusiones.....	38
5.2 Recomendaciones.....	39
6. Bibliografía	41

Lista de figuras

	<u>Pág.</u>
Fig. 1 Número de documentos por año, la oscuridad del color indica la cantidad de citas para el correspondiente año.	20
Fig. 2 Citaciones globales por año, la oscuridad del color indica el número de documentos publicados para dicho año.	20
Fig. 3. Porcentaje de autores en función del número de artículos escritos	23
Fig. 4 Colaboración entre autores.....	25
Fig. 5 Publicaciones por países.....	25
Fig. 6 Palabras clave por país.	26
Fig. 7 Evolución de los documentos publicados.	39

Lista de tablas

	<u>Pág.</u>
Tabla 1. Distribución de revistas de acuerdo a Ley Bradford.....	21
Tabla 2. Revistas más importantes.....	22
Tabla 3. Autores más importantes.....	24
Tabla 4: Artículos más citados	30
Tabla 5. Áreas temáticas.....	32

1. Introducción

En esta propuesta de trabajo final de maestría se pretende captar el interés de las comunidades académicas y profesionales, con el desarrollo de la revisión sistemática de la literatura sobre las tecnologías Docker. Esta busca destacar las contribuciones que se han producido alrededor de su funcionamiento, adaptaciones y las principales ventajas que ofrece para las áreas de la tecnología y el desarrollo computacional.

Las tecnologías Docker se han convertido en una plataforma popular y novedosa de virtualización. Esta tecnología introdujo lo que se convertiría en un estándar de la industria para contenedores, al entregar flexibilidad e innovación en este dominio. Los contenedores Docker ofrecen una respuesta al problema típico "solo funciona en mi máquina" [1], lo que proporciona muchas ventajas de portabilidad y escalabilidad para los procesos de desarrollo.

Dentro los principales usos de esta tecnología se identifican los procesos de computación en la nube y el desarrollo de aplicaciones, las cuales necesitan ser ejecutadas en entornos virtualizados, especialmente las aplicaciones de alto rendimiento computacional [2], las cuales involucran un grado de complejidad superior para los desarrolladores, por el manejo de diferentes lenguajes, estructuras de arquitectura y ambientes de trabajo.

Otro aspecto que se destaca para esta tecnología, son las ventajas que se producen en comparación con las máquinas virtuales o VM como sus siglas en inglés. Esto se explica porque los contenedores Docker virtualizan el sistema operativo y permiten desarrollar procesos de espacio aislado, logrando compartir un único Kernel (el núcleo) para ejecutar múltiples instancias; en cambio, las máquinas virtuales virtualizan consumiendo recursos del hardware que la contiene, por eso sus procesamientos pueden consumir más tiempo; de hecho, según algunos estudios precisan como las máquinas virtuales tardan algunos minutos para iniciarse, a diferencia de los contenedores Docker [3], los cuales comprenden un tiempo de respuesta en segundos; además, abarcan un consumo inferior de

almacenamiento [4]; inclusive, se le atribuye a los contenedores Docker, un rendimiento casi nativo en relación a los recursos relacionadas como memoria, CPU, disco y red [5].

Los contenedores Docker logran mantener el aislamiento a nivel de proceso de red y sistema de archivos [4], permitiendo a los desarrolladores construir múltiples contenedores replicando ambientes de producción, los cuales pueden ser construidos de manera rápida, logrando una reducción de costos para las organizaciones y el desarrollo de infraestructuras en nube [1] [5] [6].

Finalmente, las tecnologías Docker han sido adoptadas para la contenerización de aplicaciones [7], proporcionando una API de alto nivel al entregar una sobrecarga controlada en comparación con los contenedores LXC (Linux Container) [8]; de hecho, los Docker fueron construidos originalmente con LXC[9]; sin embargo, Docker escribió su propio proceso de ejecución llamado libcontainer, y posteriormente surgió otra alternativa con Ubuntu LXD, lo que comprendió un complemento para los contenedores Linux agregando nuevas funcionalidades. Concretamente, los LXD son un administrador para los contenedores y ofrecen una experiencia similar a las VM, pero en cambio hacen uso de los contenedores Linux [10][11].

Esta propuesta de trabajo final busca reunir las experiencias e investigaciones reportadas en la literatura, rescatando la información más relevante sobre las tecnologías Docker, entre los años 2014 a 2021, con el fin de identificar los principales desarrollos que se han generado alrededor del mundo, con el estudio de estas tecnologías y la virtualización basada en contenedores. Además, busca destacar los principales contextos de aplicación y usabilidad mediante el análisis de artículos publicados en las áreas de la computación y tecnologías emergentes.

El objetivo del trabajo busca responder la siguiente pregunta de investigación: ¿Cuáles son las áreas temáticas de investigación sobre tecnologías Docker y como se caracterizan? La respuesta a la pregunta anterior permitirá complementar y fortalecer las exploraciones en el campo académico, profesional y diferentes grupos de interesados como comunidades de desarrolladores, que encuentren en esta investigación información notable para la aplicación de tecnologías de contenedores, o empresas que deseen conocer el proceso de orquestación de aplicaciones en la nube, con una tecnología moderna, estándar y portátil que ha logrado posicionar un liderazgo industrial.

1.1 Protocolo de Investigación

A continuación, se describe el protocolo de investigación utilizado para desarrollar este trabajo final. La metodología de revisión sistemática consta de las siguientes fases:

- Preparación del protocolo de investigación: Corresponde al diseño de un documento donde se crea la justificación de la revisión, las preguntas de investigación, su motivación y su proceso de ejecución.
- Recopilación de los datos: Determina los documentos encontrados para el desarrollo de la revisión sistemática de la literatura.
- Procesamiento de la información recolectada: Corresponde a la selección de estudios, extracción de datos y evaluación de la calidad de artículos.
- Análisis de los resultados: Responde a las preguntas de investigación.

A continuación, se describe de forma detalla el protocolo de búsqueda detallado.

- Base bibliográfica: Scopus. Se seleccionó esta base, ya que incluye más de 219 mil títulos de diferentes editoriales tales como IEEE, Elsevier, Emerald, Springer y Wiley [12]; además, cuenta con la cobertura de 16.500 revistas revisadas por pares de las áreas de ciencias, tecnología, medicina y ciencias sociales; también, ofrece ventajas de agrupación por sus revistas indexadas.
- Ecuación de búsqueda:


```
TITLE ( docker* AND container* ) OR KEY ( docker* AND container* )
OR ABS ( docker* AND container* ) AND ( EXCLUDE ( PUBYEAR ,
1997 ) OR EXCLUDE ( PUBYEAR , 1989 ) ) AND ( LIMIT-TO (
LANGUAGE , "English" ) OR EXCLUDE ( LANGUAGE , "French" ) )
AND ( EXCLUDE ( PUBYEAR , 1987 ) OR EXCLUDE ( PUBYEAR ,
1986 ) ) AND ( EXCLUDE ( PREFNAMEUID , "Undefined#Undefined"
) )
```
- Periodo de análisis: Se utilizará toda la información disponible desde enero de 2014 hasta diciembre de 2021. Esta restricción está justificada en que dicha tecnología fue expuesta a partir el año 2013.
- Áreas temáticas: No se restringió la búsqueda por áreas temáticas.
- Se recolectaron los siguientes campos entregados por Scopus:
 - Authors.
 - Author(s) ID.

- Title.
 - Year.
 - Source title.
 - Cited by.
 - DOI.
 - Affiliations.
 - Authors with affiliations.
 - Abstract.
 - Author Keywords.
 - Index Keywords.
 - References.
 - Abbreviated Source Title.
- Criterios de inclusión y exclusión: No se aplicaron criterios adicionales de inclusión y exclusión.

La información recolectada fue sometida a un proceso de limpieza y transformación de los datos con el fin de construir nuevos campos que no entrega Scopus y unificar y formatear las columnas existentes. El proceso realizado fue el siguiente:

- Remoción de acentos.
- Cambio de formato de los nombres de los autores.
- Eliminación de la parte de los títulos que está en un idioma diferente al principal.
- Extracción de los nombres de los países de afiliación de los autores.
- Extracción de los nombres de las instituciones de afiliación de los autores.
- Limpieza y transformación de las palabras clave del autor.
- Limpieza y transformación de las palabras clave del índice.
- Transformación de las cadenas de texto a letra minúscula.

En el proceso de limpieza de palabras clave, se procedió a reemplazar por una misma cadena las variaciones que son obtenidas, por ejemplo, al considerar el plural y el singular de una misma palabra, o variaciones del dialecto como la ortografía en inglés británico y americano.

1.2 Hipótesis

Es posible determinar las áreas temáticas de investigación sobre tecnologías Docker y caracterizarlas utilizando análisis bibliométrico y de redes.

1.3 Objetivos

1.3.1 Objetivo General

Responder la siguiente pregunta de investigación: ¿Cuáles son las áreas temáticas de investigación sobre tecnologías Docker y como se caracterizan?

1.3.2 Objetivos Específicos

Los objetivos específicos de este trabajo son:

- Realizar la selección de artículos.
- Realizar el análisis bibliométrico de los artículos seleccionados.
- Responder la pregunta de investigación.
- Preparar el informe final.

1.4 Mapa del documento

Este documento se encuentra organizado de la siguiente forma: En el Capítulo 2 se presenta un corto resumen de la usabilidad de tecnologías Docker y conceptos de interés; en el Capítulo 3, se discuten los resultados obtenidos; en el Capítulo 4 se presenta la determinación y caracterización de las áreas temáticas; y en el Capítulo 5 se concluye.

2. Conceptos Básicos

Los Docker son tecnologías desarrolladas hacia el año 2013 introduciendo un estándar en la industria de la computación para el uso y desarrollo de contenedores. Esta tecnología proporciona las bases necesarias para ejecutar aplicaciones compuestas en la nube, permitiendo que estos sean administrados desde un solo anfitrión y evolucionando el método de virtualización más liviano para todo el sistema operativo[11] .

2.1 Docker

Docker es una forma de aislar un proceso del sistema en el que se está ejecutando. Permite aislar el código escrito para definir una aplicación y los recursos necesarios para ejecutar esa aplicación desde el hardware en el que se ejecuta. Tiene además como ventaja asegurar que el desarrollo local será idéntico a cualquier otro entorno en el que se desplegaría [13]. Esta tecnología se ejecuta para Linux, macOS y Windows [5][11].

2.2 Máquina Virtual

Las máquinas virtuales simulan un sistema real de computación con su propia CPU, memoria, interfaz de red y almacenamiento [2]. Estos vienen compuestos por un conjunto de archivos los cuales, en su mayoría, se pueden manipular como si se tratara de cualquier otro archivo presente en algún sistema operativo [13] [14].

Las máquinas virtuales virtualizan la aplicación y todas las librerías y dependencias, incluyendo el sistema operativo lo que demanda mayor tiempo arranque en el servidor [15].

2.3 Contenedor

Los contenedores ofrecen un camino diferente de virtualización y en la forma como se construyen las aplicaciones ya que permiten el aislamiento de recursos, virtualizando el

sistema operativo y provocando un menor consumo de recursos para ejecutar una aplicación. Particularmente las tecnologías Docker conservan un proceso exclusivo para la creación y exposición de imágenes a través de sus repositorios [16], además, los contenedores Docker tienen la propiedad de ejecutar varias instancias de una imagen permitiendo iniciarlas, detenerlas o conectarlas a otros contenedores mediante redes.

2.4 Imagen de Docker

Los contenedores Docker contienen imágenes (es el software) y con estas se puede resolver o generar una acción determinada como crearla, iniciar, parar, reiniciar o destruirla. Una imagen es una plantilla que los contenedores Docker utilizan para crear su contenedor en ejecución y son diseñadas para múltiples propósitos. Estas contienen una capa delgada de bibliotecas y binarios que son necesarias para que las aplicaciones funcionen [1]. Para definir una imagen, se debe crear un Dockerfile. Cuando estas son ejecutadas se crea el contenedor de Docker, la cual se pueden almacenar en el anfitrión, se puede eliminar o también almacenar en los repositorios [13][17]. Existe un registro de múltiples imágenes de acceso público llamado Docker Hub usado para compartir aplicaciones y automatizar flujos de trabajo [18].

2.5 Kubernetes

Kubernetes es una tecnología diseñada originalmente por Google. Son un sistema de código libre para la administración de clúster de computación, logrando que la gestión de las cargas de trabajo sea acorde a las declaraciones de los usuarios. Estos son usados para automatizar las fases de despliegue y escalado de aplicaciones en contenedores [5][8][14].

2.6 Docker Hub

Docker Hub es un repositorio público en la nube el cual distribuye contenidos y se publican multitud de imágenes de carácter gratuitos. Este es un servicio entregado por Docker para buscar y compartir imágenes de contenedor. Además, contiene imágenes para las principales tecnologías de código abierto, entre ella se encuentran: Jupyter, PostgreSQL, Redis y MongoDB [13].

2.7 Dockerfile

Es un script que guía la manera como se construye la imagen de Docker. Esta es usada con una sintaxis específica que contiene todos los comandos necesarios para ensamblar la imagen la cual le explica al Demon Docker cómo construirla [5][15][7].

2.8 Docker Client (Cliente de Docker)

Es la forma natural como el usuario interactúa con el Docker, y comprende una interfaz que contiene las instrucciones que se entregan al Docker, este se comunica con el Docker Engine (motor) el cual realiza el trabajo de contenerización. El Docker Cliente puede comunicarse con más de un Docker daemon el cual ejecuta la imagen.

2.9 Docker Engine

Es el proceso que administra los contenedores y es ocasionalmente considerando como el Docker daemon y realiza el trabajo principal de construir, ejecutar y distribuir los contenedores [13].

2.10 Docker Daemon

Lee las peticiones a través de la API del Docker y administra los objetos, como las imágenes, contenedores, redes y este tiene la capacidad de comunicarse con otros daemon para administrar los servicios de Docker [13].

3. Resultados Obtenidos

A continuación, se describen los principales resultados obtenidos.

3.1 Estadísticas básicas

Al aplicar la cadena de búsqueda presentada en el Capítulo 1 se recuperan automáticamente 1.681 documentos publicados durante los años 2014 a 2021, publicados en 884 revistas, de los cuales los artículos de conferencia comprenden la forma más popular de publicación con un 71% de participación, seguido de artículos y revisiones de conferencia con un 25% y 2% respectivamente; finalmente el 1% son capítulos de libros. Durante los años 2016 a 2020 se presentó una tasa de crecimiento media del 47%. Estos documentos fueron escritos por un total de 5.534 autores afiliados a 1.399 instituciones de diferentes países; además, estos documentos tienen en promedio 3,29 autores y tan solo 80 documentos (5%) han sido escritos por un solo autor.

A nivel general, el promedio de citas por documento es de 8,28 y el promedio de citas por documento por año es de 0,92; así mismo, en la Fig.1 se evidencia un claro crecimiento de la producción de documentos durante los años 2015 a 2020, y en la cantidad de citas el pico más alto se presenta para el año 2017 como lo demuestra la Fig. 2, siendo este el periodo donde se obtuvieron las mayores citas globales por año, con un total 3.664. Cabe anotar, que los años anteriores han sido periodos de mayor interés puesto que la intensidad del color (el promedio de citas por año) ha sido mucho más relevante, de hecho, las tres primeras publicaciones más citadas se ubican durante los años 2014 a 2016.

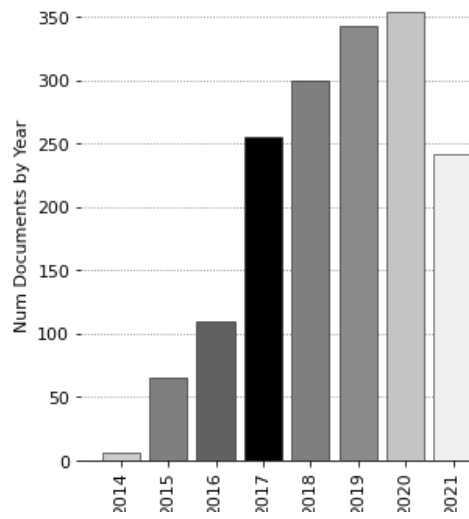


Fig. 1 Número de documentos por año, la oscuridad del color indica la cantidad de citas para el correspondiente año.

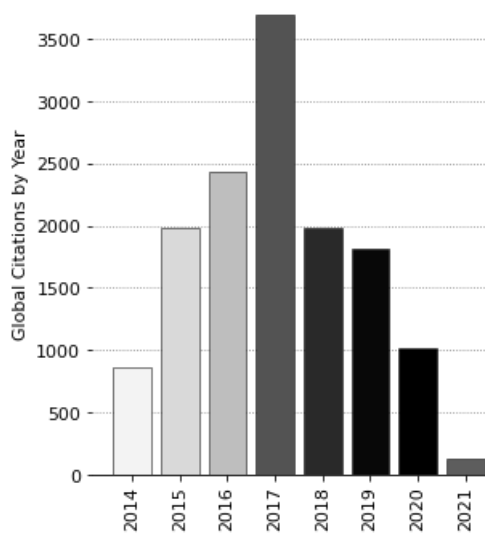


Fig. 2 Citaciones globales por año, la oscuridad del color indica el número de documentos publicados para dicho año.

3.2 Principales Revistas

La ley de Bradford permite evaluar la productividad científica y demuestra que una gran población de artículos se concentra en una pequeña población de revistas especializadas, y por tanto permite identificar las principales revistas donde se realizan las publicaciones [19]. En este trabajo se encontró que solo 33 (3,83%) revistas corresponden al foco central con 546 (32,50%) documentos publicados; y que el índice de transitoriedad es de 663 (75%); esto implica que hay 663 revistas que han realizado una sola publicación; esto se puede identificar en la Tabla 1.

En la Tabla 2 se presentan las 10 revistas con más citaciones globales; además, se anexan los índices de impacto que permiten evaluar la importancia de la revista de acuerdo al número de publicaciones y número de citaciones por publicación [20]. Las revistas IEEE International Conference on Cloud Computing y la revista IEEE Access destacan por la exposición de artículos que han logrado el mayor número de citaciones globales las cuales se concentran en publicaciones acerca de contenedores Docker, temas de Edge computing e IoT; de hecho, muchos de los artículos con mayor número de citaciones se reúnen en estas revistas. Estos fueron descritos en la sección de artículos más importantes.

Tabla 1. Distribución de revistas de acuerdo a Ley Bradford.

Num. Revistas	% Acum	Documentos Publicados	% Publicaciones	Grupo Bradford
34	3,85%	546	32,50%	1
187	21,15%	471	28,04%	2
663	75,00%	663	39,46%	3

Tabla 2. Revistas más importantes

Revista	Número de documentos	Total citas	Citaciones Globales por año	Promedio Global de Citaciones	Índice H	Índice M	Índice G
Lecture Notes in Computer Science	78	406	50.75	5.21	11.00	1.38	4
Journal of Physics: Conference Series	42	36	4.50	0.86	3.00	0.38	2
ACM International Conference Proceeding	38	213	23.67	5.61	6.00	0.67	3
Communications in Computer and Information Science	36	60	8.57	1.67	5.00	0.71	2
CEUR Workshop Proceedings	36	52	6.50	1.44	4.00	0.50	2
Advances in Intelligent Systems and Computing	27	42	6.00	1.56	4.00	0.57	2
PervasiveHealth: Pervasive Computing Technologies for Healthcare	26	36	9.00	1.38	4.00	1.00	2
IEEE International Conference on Cloud Computing, CLOUD	23	313	52.17	13.61	10.00	1.67	5
IEEE Access	21	210	35.00	10.00	6.00	1.00	3
GigaScience	20	179	22.38	8.95	8.00	1.00	4

3.3 Principales Autores

La base datos está conformada por un total de 5.534 autores; de ellos, el 16,9% ha publicado el 51,8% del total de los artículos recolectados. En la Fig.3 se presenta el número de documentos publicados respecto al número de autores; allí se puede apreciar como una minoría de autores publican la mayor parte de artículos relevantes sobre el campo de investigación; este fenómeno es explicado de acuerdo a la ley bibliométrica anunciada por Alfred Lotka (Ley de Lotka), quien define que la mayor producción de artículos se concentra en una minoría de autores altamente productivos y a su vez un número de muchos autores producen una única publicación [21].

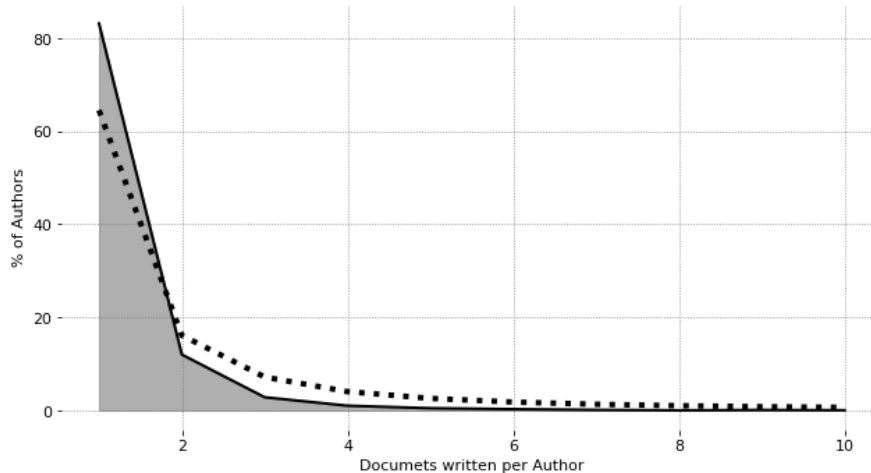


Fig. 3. Porcentaje de autores en función del número de artículos escritos

En la Tabla 3 se presenta los autores con mayor número de publicaciones; además, de sus indicadores de impacto en la producción de artículos; Fetzer tiene un total de 5 artículos y 438 citas; sus contribuciones se asocian con orquestación de contenedores [22]; entre sus trabajos se destacan: Intel SGX (Software Guard Extensions) un soporte para Kubernetes, en esta publicación se presentan detalles sobre su proceso de arquitectura e implementación [22]; GENPACK, que es un marco de programación de contenedores para la asignación de recursos en centros de datos en las nubes partiendo del principio de desperdicio generacional (GC) [23], entre otros estudios. La más citada de sus publicaciones se da en colaboración con Arnautov et. al 2016, donde se aborda un mecanismo de contenedor seguro (SCONE) haciendo uso de Intel SGX (Software Guard Extensions); estos códigos proveen mecanismos de seguridad en los contenedores Linux y por supuesto son mecanismos compatibles con Docker, presentando un impacto mínimo en su rendimiento [24]. Las contribuciones de los principales autores están descritas en la sección de artículos más citados.

Tabla 3. Autores más importantes.

Autor	No. Publicaciones	Citaciones totales	Frac. Num. Documentos	Citaciones globales Por año	Promedio Citaciones globales	Índice H	Índice M	Índice G
Fetzer C	5	438	0.69	62.57	87.6	3	0.43	3
Stillwell M	2	418	0.57	52.25	209	2	0.25	2
Gregor F	2	402	0.18	57.43	201	2	0.29	1
Martin A	2	400	0.24	57.14	200	1	0.14	1
Felter W	1	553	0.25	69.12	553	1	0.12	1
Ferreira A	1	553	0.25	69.12	553	1	0.12	1
Rajamony R	1	553	0.25	69.12	553	1	0.12	1
Rubio J	1	553	0.25	69.12	553	1	0.12	1
Bernstein D	1	550	1.00	61.11	550	1	0.11	1
Arnautov S	1	399	0.07	57.00	399	1	0.14	1

Dentro de la colaboración entre autores se identifica en la Fig. 4 un top diez sobre la correlación de autorías donde haya sucedido por lo menos más de una publicación que tenga asociación con el título del artículo; el tamaño y color de cada nodo indica el grado de correlación; además, el índice de correlación fue calculado usando el método Pearson. Fetzer, como uno de los principales autores tiene un mayor grado asociación de publicación de artículos con Gregor y Martin en [24]; en publicaciones con una correlación menos fuerte, presenta correlación en [25] y [26]; también; en otras colaboraciones se identifica correlación con Dua en las publicaciones [27], [28] y con Stillwell en [24] y [29].

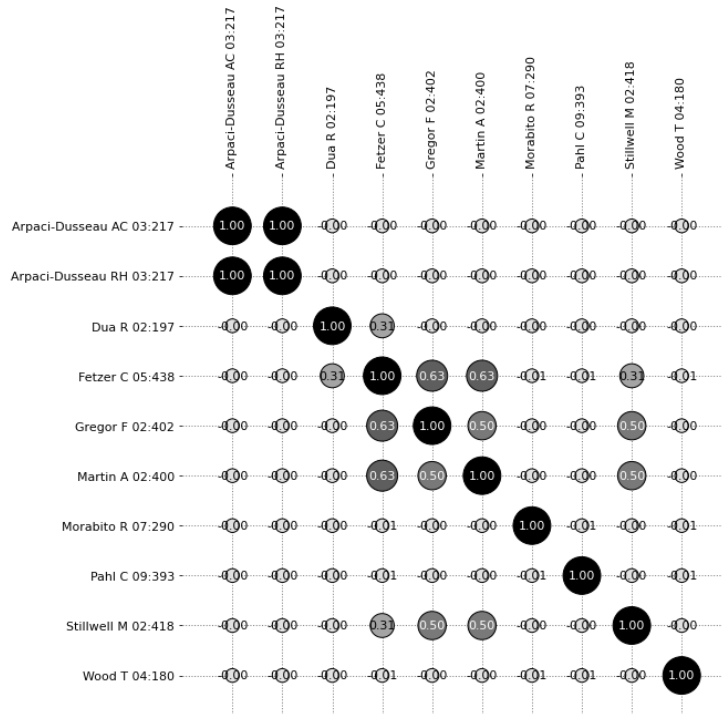


Fig. 4 Colaboración entre autores.

3.4 Publicaciones por país

Se encontró que la mayor publicación a nivel de países se concentra en: Estados Unidos (18%), China (11%), Alemania (7%), India (6%), Italia (6%) y Reino Unido (5%). Finalmente, 13 países (18%) tienen una única publicación. En la Fig. 5 se presenta un mapa con la cantidad de documentos por país.



Fig. 5 Publicaciones por países.

En la Fig. 6 se observa la concentración de términos clave por país; los números que siguen al nombre de cada país o palabra clave, indican la cantidad de documentos y la cantidad de citas, la oscuridad del color se da por el grado de ocurrencia de artículos con estas palabras claves; por ejemplo, la palabra kubernetes tiene un mayor grado de investigación en países como Estados Unidos, China y la India; estos países se caracterizan por ser potencias en el desarrollo tecnológico y creación de software; en esta misma dirección, las palabras Big data, Machine Learning y microservicios, también son analizados especialmente por países como Estados Unidos, China, India y Canadá que también tiene una frecuencia similar para los artículos con contenidos de microservicios. El tema de seguridad es estudiado principalmente por Estados Unidos y Reino Unido, y los temas de IoT son trabajados especialmente por países como India y Estados Unidos.

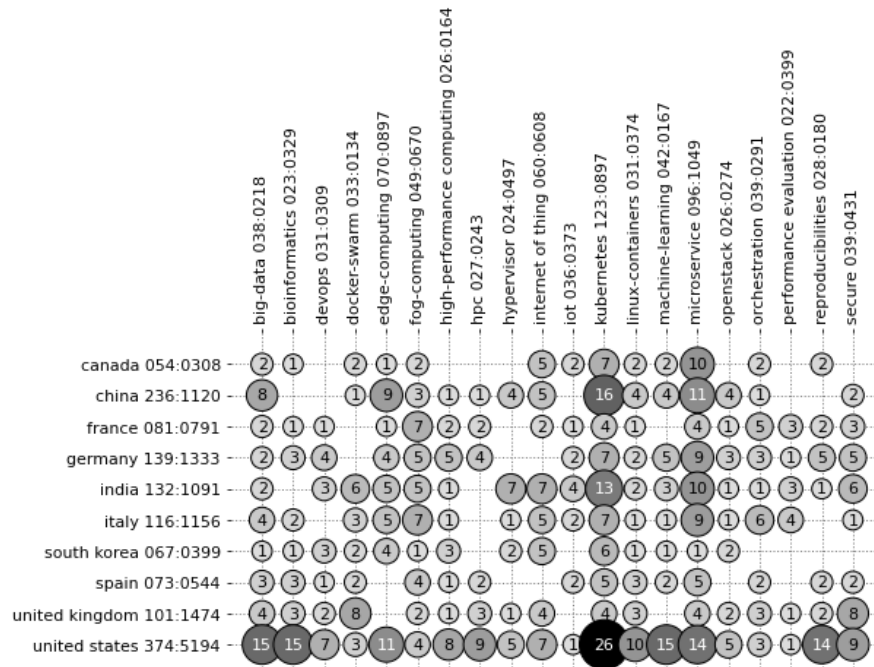


Fig. 6 Palabras clave por país.

3.5 Artículos más citados

En la Tabla 4 se presentan los 10 documentos más importantes; el artículo más citado fue el trabajo de Felter et al. 2015 [30] en el cual se compara el rendimiento

de la virtualización entre máquinas virtuales y contenedores al ejecutar las aplicaciones Redis y MySQL. El trabajo concluye que los contenedores Docker son una tecnología viable a escala de un servidor completo, ya que esta tecnología tiene un impacto mínimo en el rendimiento [30].

En Bernstein (2014) [31] se discute la diferencia de llevar un sistema a la nube desde un hipervisor o un contenedor, y resalta como los contenedores comparten un único sistema operativo y esto lo convierte en una implementación de menor tamaño, lo que hace posible almacenar cientos de contenedores en un host físico; además, se aclara en un detalle amplio el funcionamiento de contenedores Docker y como estos permiten un grado de automatización más rápido para aplicaciones Linux.

En (Arnautov, 2016) [24] se ofrece una mirada sobre aspectos de seguridad de los contenedores, habla acerca de las vulnerabilidades de aislamiento, la cual es una condición que compromete la seguridad de los contenedores. Esta publicación además se centra sobre SCONE, este es un medio de soporte para proteger ataques externos, lo cual permite ofrecer una condición de seguridad sobre la confidencialidad e integridad de los datos en partes no autorizadas.

En la referencia [27] se aclara los retos que tienen los proveedores de PaaS (plataforma como servicios) para proporcionar de una manera más eficiente servicios de computación en la nube; en este caso, los autores concentran un estudio en un ecosistema de diferentes tecnologías de contenedores como: LXC, Warden, Docker y OpenVZ, en los cuales destacan sus principales diferencias, características técnicas, entornos en los cuales pueden desarrollarse, limitaciones de seguridad y una ficha técnica para definir cuál es el tipo de contenedor que es más recomendado usar.

En (Ismail, 2016) [32] se presenta un estudio que explica como Docker tiene capacidades tecnológicas que lo convierten en una plataforma Edge computing viable, ya que son una alternativa más ligera, portátil y de alto rendimiento lo cual lo hace adecuado para iniciar aplicaciones Edge mucho más eficiente que una

máquina virtual. Docker contribuye como un adecuado candidato por sus capacidades de integración y despliegue continuo.

En la referencia [33] se presenta un trabajo acerca de OpenNetVM, una plataforma de alto rendimiento la cual se basa en contenedores Docker, usando un marco de procesamiento que simplifica el desarrollo de las funciones de red así como su gestión y optimización; se aclara como las funciones de red se ejecutan como procesos de espacio de usuario en los contenedores Docker, lo que lo hace más liviano que las VM, simplificando la utilización de las NFs (Network Functions) en múltiples proveedores; además, destaca que cada contenedor puede poseer sus propias librerías y dependencias, lo cual permite conservar una característica de aislamiento.

El trabajo de Stanciu A., explora el mundo del Edge Computing, el cual comprende un novedoso modelo para proveer recursos computacionales cerca de dispositivos conectados a la red (IOT, redes inteligentes, entre otros); su trabajo asocia una investigación sobre la tecnología blockchain como plataforma de sistemas de control jerárquico y distribuido. La integración con el Edge Computing se basa en una arquitectura de microservicios donde los contenedores Docker implementan bloques de funciones y kubernetes se ejecuta para orquestar la ejecución de los contenedores [34] [35].

Otro tema que se expone en la literatura se enfoca en entender las condiciones presentes para migrar arquitecturas de microservicios; en el artículo publicado por Taibi et al. 2017 [36], se realizan encuestas con algunos profesionales como ingenieros de software, gerentes de proyectos, desarrolladores, entre otros, quienes tienen experiencia realizando procesos de migración; en principio, se logra concluir como muchas empresas lo ven como una solución viable, otras lo asumen como un cambio que deben seguir ya que es una tendencia que grandes y pequeñas empresas lo están aplicando, tales como Amazon, Netflix, LinkedIn y muchos otras más. En otro sentido, las encuestas también permiten identificar un rechazo por el costo inicial que provoca migrar arquitecturas a microservicios; una

de las razones es porque no se identifica rápidamente el retorno de la inversión que se requiere para adoptar este tipo de arquitecturas, especialmente, por la complejidad que se da al inicio para introducir un proceso de desacoplamiento de estructuras monolíticas, un esfuerzo que además provoca adoptar una infraestructura DevOps, entre otros; también, se resaltan otros aspectos que interfieren como lo son temas legales y amenazas de seguridad [36].

Finalmente, en [37] se observa cómo se analizan diferentes alternativas para atender las limitantes que existen para los sistemas de salud modernos, utilizando tecnologías modernas como blockchain y contenedores Docker que ayuden a mejorar el acceso, eficiencia y seguridad en el manejo de los datos y que permitan simular entornos para implementar sistemas EHR (registro de salud electrónico) ayudando a mejorar el control y flujo de información entre los proveedores del sistema médico.

Tabla 4: Artículos más citados

Artículo	Revista	Citaciones Normalizadas
Felter et al. 2015. An updated performance comparison of virtual machines and Linux containers.	ISPASS 2015 - IEEE International Symposium on Performance Analysis of Systems and Software	0.237
Bernstein D. 2014. Containers and cloud: From LXC to docker to kubernetes.	IEEE Cloud Computing	0.272
Arnautov S et al. 2016. SCONE: Secure linux containers with Intel SGX.	Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016	0.197
Pahl C. 2015. Containerization and the PaaS Cloud.	IEEE Cloud Computing	0.105
Dua R et al. 2014. Virtualization vs containerization to support PaaS.	Proceedings - 2014 IEEE International Conference on Cloud Engineering, IC2E 2014	0.093
Tanwar S. et al. 2020. Blockchain-based electronic healthcare record system for healthcare 4.0 applications.	Journal of Information Security and Applications	0.081
Ismail BI et al. 2016. Evaluation of Docker as Edge computing platform.	ICOS 2015 - 2015 IEEE Conference on Open Systems	0.080
Stanciu A. 2017. Blockchain Based Distributed Control System for Edge Computing.	Proceedings - 2017 21st International Conference on Control Systems and Computer, CSCS 2017	0.065
Zhang W(4) et al. 2016. OpenNetVM: A platform for high performance network service chains.	HotMiddlebox 2016 - Proceedings of the 2016 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization, Part of SIGCOMM 2016	0.063
Taibi D, et al. 2017. Processes, Motivations, and Issues for Migrating to Microservices Architectures	IEEE Cloud Computing	0.058

4. Determinación y caracterización de las áreas temáticas sobre tecnologías Docker.

4.1 Determinación de las áreas temáticas.

Sobre la base de 1681 publicaciones de investigación de tecnologías Docker durante los años 2014 a 2021 se realiza el análisis de temas de interés con el objetivo de identificar los principales áreas temáticas y evolución en el campo de investigación. Esta selección se lleva a cabo con todos los artículos en idioma inglés y sometiendo las palabras claves del autor en un proceso de limpieza automático en el que se desarrollaron algunos aspectos de calidad inicial; tales como, eliminación de automática mayúsculas, signos y caracteres especiales, haciendo uso de un tesoro; posteriormente, se ejecutó un proceso de limpieza manual en el que se excluyeron las palabras relacionadas con la ecuación de búsqueda, con el fin de incluir en el análisis las palabras no obvias del campo; finalmente, se hace uso a través de una aplicación en Python el desarrollo de clusterización, la cual emplea algoritmo de Louvain donde se definen como parámetros palabras claves que aparecen con un mínimo de ocurrencia en 10 documentos y haciendo uso del índice de asociación para normalizar, lo cual ofrece como resultado un total de 4 clúster representados por los términos de mayor frecuencia.

A continuación, se presenta el resultado en la Tabla 5.

Tabla 5. Áreas temáticas

Nombre del clúster	Términos más frecuentes
HPC	big data, machine learning, bioinformatics, workflow, reproducibilities, singularity, deep learning, container technologies, cluster, hadoop, pipeline, distributed computations, web service, anomaly detection, gpu.
internet of thing	internet of thing, edge computation, fog computing, network function virtualization, software defined network, 5g, kvm, scalability, blockchain, managing resources, deployment, live migration, raspberry.
Microservice	microservice, cloud, orchestration, secure, devops, hypervisor, performance evaluation, monitoring, platform as a service, distribute systems, unikernel, toasca.
Kubernetes	kubernetes, docker swarm, openstack, load balance, quality of service, auto scaling, network, energy efficiency, aws.

4.2 Caracterización de los clústeres

4.2.1 Clúster 1

El Clúster 1 está relacionado con las aplicaciones de la computación de alto desempeño (HPC) implementadas en un contenedor; estas tienen que ver con la ejecución de modelos de machine learning y deep learning, aplicaciones de Hadoop y de los demás temas relacionados con el cómputo distribuido y servicios web [38]. El uso de contenedores expone los beneficios de la virtualización de servicios HPC, y se demuestra la eficiencia del rendimiento usando tecnologías Docker; por ejemplo, en [38], [39], [40], se desarrollan escenarios y experimentos sobre computación en la nube que demuestran el rendimiento usando estas tecnologías.

Dentro de esta misma área temática, se incluyen nuevas tecnologías que desean brindar mayor seguridad y rendimiento en los procesos de virtualización para sistemas de HPC. Azab [41] desarrolla Socker que es una aplicación para ejecutar contenedores Docker de forma segura; en este trabajo el autor demuestra que no se incurre en costos computacionales adicionales a los introducidos por el motor de Docker. Por otra parte, Singularity es un nuevo enfoque de contenerización que se enfoca en los problemas únicos de los sistemas HPC; sin embargo, Docker presenta algunas bondades sobre Singularity para su uso en la nube [39]. No obstante, hay autores (como en [33] y [39]) que afirman que Docker tiene un impacto aún incierto; esta conclusión se basa en un experimento de escalado de aplicaciones de HPC, haciendo uso de supercomputadoras CrayXC30 y contenedores Docker en Amazon Elastic Compute Cloud; al final, se logra concluir que el escalado de aplicaciones se adapta mejor a recursos de supercomputadoras en relación a los contenedores Docker.

Los trabajos más destacables en esta área son desarrollados por los autores: Chung [38], Quang-Hung [38], Nguyen [38], Thoai [38], Younge [39], Pedretti [39],

Grant [39], Brightwell [39], Azab [42], Ruiz [40], Jeanvoine [40], Nussbaum [40], Sparks J [41].

4.2.2 Clúster 2

En el Clúster 2 se aborda el tema del IoT (Internet of things); este clúster está relacionado con los temas de computación de borde, computación en la niebla, la virtualización, la transmisión de información a través de redes 5g y servicios de blockchain. El concepto de IoT surge por la gran cantidad de dispositivos conectados a internet para diferentes propósitos de uso común; dicha cantidad de dispositivos ha crecido rápidamente y ellos generan una enorme cantidad de datos donde se hace necesario proveer recursos computacionales eficientes [43]; justamente, una de las relaciones con las tecnologías Docker es lograr combinar los datos producidos por dispositivos conectados a internet con métodos de virtualización que permitan reducir costos de transmisión y procesamiento de información en la nube; además, estas tecnologías deben proporcionar capacidades de recuperación, aislamiento y almacenamiento de datos de los dispositivos [43]–[50]; también, las plataformas en la nube modernas a gran escala requieren una técnica de migración en vivo en contenedores Docker que permita soportar el equilibrio de carga, el mantenimiento del host y la mejora de la calidad del servicio (QoS) [51]. Algo semejante ocurre con el concepto de blockchain, esta tecnología es usada como plataforma de control distribuido, ofreciendo un estado superior de seguridad y confianza sobre los datos producidos por los dispositivos conectados a internet. Un ejemplo, es el trabajo desarrollado en [35], donde se utilizó el proyecto de Hyperledger Fabric, el cual ofrece un marco de código abierto que se integra con los contenedores Docker como bloque de función principal para la construcción de aplicaciones; también, se hace uso de kubernetes como plataforma para administrar y orquestar la ejecución de los contenedores en los recursos a borde; al final, logra demostrarse un esquema exitoso para proporcionar recursos computacionales cerca de los dispositivos finales que están conectados a la red [44]. Otros estudios como los realizados por Morabito et. al [52], destacan las ventajas de aprovechar al máximo la posibilidad de usar las tecnologías Docker

en los dispositivos de placa única (SBC), exponiendo las fortalezas y debilidades presentes en dispositivos de bajo consumo (los cuales se caracterizan por tener menos recursos computacionales), al manejar instancias virtualizadas en contenedores [53]; ellos concluyen que Docker genera un impacto mínimo en términos de rendimiento versus ejecuciones nativas; sin embargo, en el caso de la computación en la niebla, si bien usar procesos de contenerización permite tener ventajas de escabilidad y optimización de recursos [32], en [54] [55] se explica que lograr este resultado dependerá de adecuadas técnicas de implementación, ya que en este escenario de plataformas la descarga de imágenes de contenedores suele ser muy lenta, esto porque al descargar y descomprimir las imágenes de Docker consume un tiempo significativo que impactan diferentes elementos de hardware como lo son el ancho de banda, CPU, E/S de manera secuencial; precisamente, uno de los desafíos más importantes en la computación de niebla es el que generalmente se usa con ordenadores de placa única como Raspberry Pis los cuales tienen una capacidad de almacenamiento limitada.

Por último, otra conclusión importante acerca de las tecnologías Docker es que existen temas que aún pueden ofrecer un mayor grado de atención en las investigaciones para evaluar las interacciones en múltiples puertos de enlace; de hecho, la misma tecnología Docker ofrece pautas para conocer y ampliar el conocimiento sobre problemas de seguridad dentro de las aplicaciones virtualizadas [53].

Los trabajos más citados en esta área son desarrollados por los autores: Stanciu [35], Morabito R. [52]- [52], Dehury [43], Sahoo [43], Ahmed [54]-[55], Ramalho [45], Neto [45], Mendki [46].

4.2.3 Clúster 3

El siguiente clúster está relacionado con la orquestación óptima de servicios y microservicios utilizando contenedores de Docker; una de las aplicaciones más representativas de este clúster es el uso de la plataforma como servicios (PaaS) cuando la plataforma es implementada en un contenedor de Docker. Los

microservicios representan un conjunto de pequeños servicios acoplados que se comunican entre sí, lo cual permite agilizar las fases de desarrollo. Desde hace algunos años se viene presentando una tendencia en la cual las industrias están adoptando la migración de aplicaciones monolíticas a versiones que permitan desarrollar servicios de forma independiente y más ágil; ante todo, trabajar en arquitecturas de microservicios promueve la reducción de complejidad y el mantenimiento de diferentes niveles de escalabilidad [36], [53]–[56].

En [58] se busca evaluar las características de las cargas de trabajo en nube; para ello, se mide el rendimiento generado en dos lenguajes popularmente utilizados Node.js y Java; el experimento consiste en evaluar su resultado en una versión monolítica y de microservicios; como resultado se concluye que la versión en microservicios puede consumir casi un 80% menos en recursos de hardware; también, en Node.js se identificó que consume un poco más de tiempo en procesar una solicitud de usuario (valoraciones a nivel de hardware y software); su relación con las tecnologías Docker surge por el hallazgo de identificar la necesidad de incorporar un proceso de virtualización que ayuda a la optimización de los tiempos de ejecución y despliegue de los microservicios ya que los tiempos medidos dentro del experimento pueden verse impactados por la virtualización de red del contenedor Docker.

Finalmente, los trabajos más citados en esta área son desarrollados por los autores: Taibi [36], Lenarduzzi [36], Pahl [36], Ueda [58], Nakaike [58], Ohara [58], Taherizadeh [59], Stankovski [59], Grobelnik [59].

4.2.4 Clúster 4

Finalmente, el clúster 4 está relacionado con el uso de Kubernetes para el balance automático de cargas, el auto escalado y la eficiencia de energía que deben tener los contenedores de Docker cuando están operando en la nube, y en el cual los contenedores van cargándose y destruyéndose de forma automática; dentro de los temas abordados se encuentra la administración de Kubernetes y Swarm, las tecnologías Openstack (las cuales también realizan procesos de virtualización con

algunas ventajas frente a las tecnologías de Docker). En general Kubernetes es una plataforma de código abierto que suministra mecanismos para monitorear dinámicamente los requisitos de recursos, el uso de las aplicaciones en ejecución, y realizar el escalamiento de aplicaciones en clúster de hosts [34], [60]–[65].

Los contenedores Docker son relevantes para las nubes de PaaS para administrar y orquestar el despliegue de aplicaciones en la nube; su funcionamiento se usa para construir, enviar y ejecutar cualquier aplicación en cualquier lugar; Swarm ayuda en la administración del Docker y Kubernetes permite administrar y el escalado de los contenedores, permitiendo una gestión más eficiente de las aplicaciones [65].

Los trabajos más citados en esta área son desarrollados por los autores:

Pahl [60], Joy [34], Dupont [61], Giaffreda [61], Capra [61], Netto [62], Lung [62], Correia [62], Luiz [62], Sa de Souza [62], Chang [63], Yang [63], Yeh [63], Lin [63], Jeng [63], Shah [65]y Dubaria [65].

5. Conclusiones y recomendaciones

5.1 Conclusiones

Se realizó el análisis bibliométrico sobre 1.681 artículos de investigación científica sobre tecnologías Docker haciendo uso de procesos de extracción, limpieza y transformación de los términos y cadenas de texto; además, se aplican metodologías basadas en el análisis de texto, coocurrencias e palabras claves de autor y diseño de clúster, de esta forma se permite valorar aspectos como el crecimiento de la producción de publicaciones en revistas, la productividad de la investigación por países, destacar los autores más importantes, los artículos más relevantes y los temas principales que circulan alrededor de este tema de investigación.

Frente a los temas más notables alrededor de las tecnologías Docker se concluye que están dominados por el análisis de cuatro temáticas importantes y relacionadas entre sí, además, que incluyen procesos de infraestructura moderna para la nube, estas son:

- Arquitectura de microservicios
- Procesos de orquestación y administración de contenedores con Kubernetes.
- Los sistemas de computación de alto rendimiento (HPC).
- El contexto naciente para el aprovechamiento, procesamiento y virtualización de la información generada desde los dispositivos IoT.
- Las tecnologías Docker ofrecen una menor huella de recursos y ofreciendo un rendimiento casi nativo y ahorros importantes en los ciclos de desarrollo e infraestructuras para la nube, han sido una solución clave para el despliegue continuo y posicionándose como una tecnología de software que lidera en la industria.

5.2 Recomendaciones

- Sería interesante realizar un análisis bibliográfico independiente para cada clúster ya que en el análisis de términos se identifica una participación de interés durante el último año para los términos claves objetos de estudio en las temáticas relevantes para tecnologías Docker, lo cual se puede apreciar en la Fig. 7 (En el eje horizontal representa el número de artículos publicados de acuerdo a la escala definida antes del año 2021 y entre el año 2021-2022).

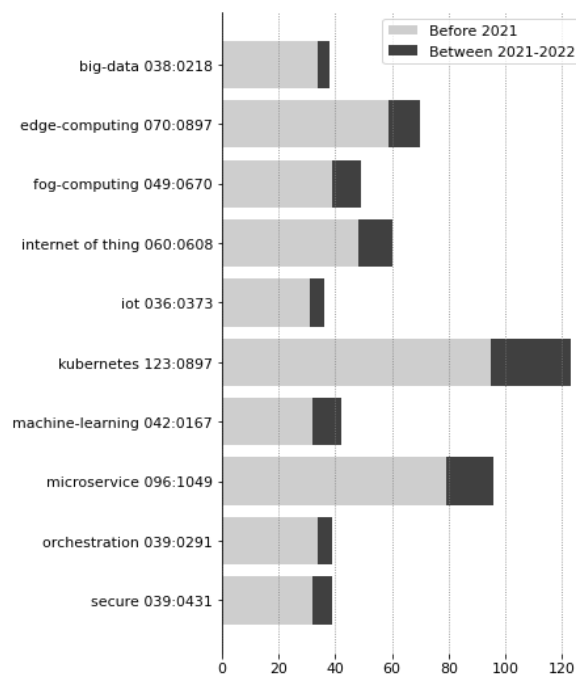


Fig. 7 Evolución de los documentos publicados.

- Las tecnologías Docker representan uno de los avances más importantes en la industria del software y la computación para el despliegue de aplicaciones; no obstante, esta tecnología sigue trabajando en algunos retos importantes, uno de ellos se da en temas de seguridad; de hecho, es un aspecto que valdría la pena explorar ya que se identifican pocas investigaciones o artículos alrededor del tema; aun así, existen algunos que señalan vulnerabilidades o características más débiles específicamente para la protección e integridad de los datos en comparación a las VM.

- Es importante recomendar seguir la investigación y lectura sobre los cambios ascendentes publicados alrededor de Docker para conocer sus nuevas integraciones con otros desarrollos de software para fortalecer condiciones de seguridad, avances en aspectos que vienen en constante actualización para procesos de aislamiento de cargas, disponibilidad de la tecnología para nuevos sistemas operativos, métodos para dockerizar aplicaciones en Python, entre otros.

6. Bibliografía

- [1] C. Anderson, "Docker," *IEEE Software*, vol. 32, no. 3, pp. 102–105, 2015, doi: 10.1109/MS.2015.62.
- [2] M. T. Chung, N. Quang-Hung, M. Nguyen, and N. Thoai, "Using Docker in high performance computing applications," in *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*, 2016, pp. 52–57, doi: 10.1109/CCE.2016.7562612.
- [3] M. L. and N. M. J. Bhimani, Z. Yang, "Accelerating big data applications using lightweight virtualization framework on enterprise cloud," *2017 IEEE High Performance Extreme Computing Conference (HPEC)*, pp. 1–7, 2017, doi: 10.1109/HPEC.2017.8091086.
- [4] R. Dua, A. R. Raja, and D. Kakadia, "Virtualization vs containerization to support PaaS," *Proceedings - 2014 IEEE International Conference on Cloud Engineering, IC2E 2014*, pp. 610–614, 2014, doi: 10.1109/IC2E.2014.41.
- [5] D. Bernstein, "Containers and cloud: From LXC to docker to kubernetes," *IEEE Cloud Computing*, vol. 1, no. 3, pp. 81–84, 2014, doi: 10.1109/MCC.2014.51.
- [6] T. Combe, A. Martin, and R. Di Pietro, "To Docker or Not to Docker: A Security Perspective," *IEEE Cloud Computing*, vol. 3, no. 5, pp. 54–62, 2016, doi: 10.1109/MCC.2016.100.
- [7] J. Cito, G. Schermann, J. E. Wittern, P. Leitner, S. Zumberi, and H. C. Gall, "An Empirical Analysis of the Docker Container Ecosystem on GitHub," *IEEE International Working Conference on Mining Software Repositories*, pp. 323–333, 2017, doi: 10.1109/MSR.2017.67.
- [8] P. Bellavista and A. Zanni, "Feasibility of fog computing deployment based on docker containerization over RaspberryPi," *ACM International Conference Proceeding Series*, 2017, doi: 10.1145/3007748.3007777.
- [9] N. Naik, "Building a virtual system of systems using docker swarm in multiple clouds," *ISSE 2016 - 2016 International Symposium on Systems Engineering - Proceedings Papers*, pp. 7–9, 2016, doi: 10.1109/SysEng.2016.7753148.
- [10] "linuxcontainers." [Online]. Available: <https://linuxcontainers.org/lxd/introduction/>

- [11] R. Peinl, F. Holzschuher, and F. Pfitzer, "Docker Cluster Management for the Cloud - Survey Results and Own Solution," *Journal of Grid Computing*, vol. 14, no. 2, pp. 265–282, 2016, doi: 10.1007/s10723-016-9366-y.
- [12] J. D. Velásquez, "A short guide to writing systematic literature reviews. Part 4," *DYNA (Colombia)*, vol. 82, no. 190. Universidad Nacional de Colombia, pp. 9–12, 2015.
- [13] M. D 'agostino, *Docker for Data Scientists*. .
- [14] A. M. Joy, "Performance comparison between Linux containers and virtual machines," in *2015 International Conference on Advances in Computer Engineering and Applications*, 2015, pp. 342–346, doi: 10.1109/ICACEA.2015.7164727.
- [15] D. Liu and L. Zhao, "The research and implementation of cloud computing platform based on docker," *2014 11th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2014*, pp. 475–478, 2014, doi: 10.1109/ICCWAMTIP.2014.7073453.
- [16] R. Peinl, F. Holzschuher, and F. Pfitzer, "Docker Cluster Management for the Cloud - Survey Results and Own Solution," *Journal of Grid Computing*, vol. 14, no. 2, pp. 265–282, 2016, doi: 10.1007/s10723-016-9366-y.
- [17] R. Morabito, "Virtualization on internet of things edge devices with container technologies: A performance evaluation," *IEEE Access*, vol. 5, no. c, pp. 8835–8850, 2017, doi: 10.1109/ACCESS.2017.2704444.
- [18] B. P. and Y. S. Preeth E N, F. J. P. Mulerickal, "Evaluation of Docker containers based on hardware utilization," *International Conference on Control Communication & Computing India (ICCC)*, pp. 697–700, 2015.
- [19] J. Ardanuy, "Breve introducción a la bibliometría," *La base de datos scopus y otros e-recursos del CBUES como instrumento de gestión de la actividad investigadora*; 1, 2012.
- [20] L. Egghe, "Theory and practise of the g-index," Springer, 2006.
- [21] R. URBIZAGASTEGUI, "La ley de Lotka y la literatura de bibliometría," *Investigación Bibliotecológica: archivonomía, bibliotecología e información*, vol. 13, no. 27, Jul. 1999, doi: 10.22201/iibi.0187358xp.1999.27.3913.
- [22] S. Vaucher, R. Pires, P. Felber, M. Pasin, V. Schiavoni, and C. Fetzer, "SGX-Aware container orchestration for heterogeneous clusters," in *Proceedings - International Conference on Distributed Computing Systems*, 2018, vol. 2018-July, pp. 730–741, doi: 10.1109/ICDCS.2018.00076 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85050967176&doi=10.1109%2fICDCS.2018.00076&partnerID=40&md5=8c379efa469c8e5d1977dff357ace5e>
- [23] A. Havet, V. Schiavoni, P. Felber, M. Colmant, R. Rouvoy, and C. Fetzer, "GENPACK: A generational scheduler for cloud data centers," in *Proceedings - 2017 IEEE International Conference on Cloud Engineering, IC2E 2017*, 2017, pp.

95–104, doi: 10.1109/IC2E.2017.15 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85020209774&doi=10.1109%2fIC2E.2017.15&partnerID=40&md5=38d3f07f32c0b6fa25d70c9254bdf66d>

- [24] S. Arnautov *et al.*, “SCONE: Secure linux containers with Intel SGX,” in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*, 2016, pp. 689–703 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85068857453&partnerID=40&md5=8b0ff57dcc20192e2eab19240e87c87b>
- [25] R. Krahn *et al.*, “TEEMon: A continuous performance monitoring framework for TEEs,” in *Proceedings of the 21st International Middleware Conference*, 2020, pp. 178–192.
- [26] R. Peterson, A. da Silva, A. Carvalho, C. Fetzer, A. Martin, and I. Blanquer, “Vallum-med: Protecting medical data in cloud environments,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 3397–3400.
- [27] R. Dua, A. R. Raja, and D. Kakadia, “Virtualization vs containerization to support PaaS,” in *Proceedings - 2014 IEEE International Conference on Cloud Engineering, IC2E 2014*, 2014, pp. 610–614, doi: 10.1109/IC2E.2014.41 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84908583822&doi=10.1109%2fIC2E.2014.41&partnerID=40&md5=d2d99e90b5614428c50244d72304fbd3>
- [28] R. Dua, V. Kohli, S. Patil, and S. Patil, “Performance analysis of Union and CoW File Systems with Docker,” in *International Conference on Computing, Analytics and Security Trends, CAST 2016*, 2017, pp. 550–555, doi: 10.1109/CAST.2016.7915029 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85019901336&doi=10.1109%2fCAST.2016.7915029&partnerID=40&md5=669f81bd5bac01919a437f6e7dfce093>
- [29] M. Stillwell and J. G. F. Coutinho, “A DevOps approach to integration of software components in an EU research project,” in *Proceedings of the 1st International Workshop on Quality-Aware DevOps*, 2015, pp. 1–6.
- [30] W. Felter, A. Ferreira, R. Rajamony, and J. Rubio, “An updated performance comparison of virtual machines and Linux containers,” in *ISPASS 2015 - IEEE International Symposium on Performance Analysis of Systems and Software*, 2015, pp. 171–172, doi: 10.1109/ISPASS.2015.7095802 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84937435024&doi=10.1109%2fISPASS.2015.7095802&partnerID=40&md5=8f6a0996b26e08c6d457ccb23c20bf57>
- [31] D. Bernstein, “Containers and cloud: From LXC to docker to kubernetes,” *IEEE Cloud Computing*, vol. 1, no. 3, pp. 81–84, 2014, doi: 10.1109/MCC.2014.51. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0->

84923248887&doi=10.1109%2fMCC.2014.51&partnerID=40&md5=a4cdae0851bad
e9e9485a070ac581c17

- [32] B. I. Ismail *et al.*, "Evaluation of Docker as Edge computing platform," in *ICOS 2015 - 2015 IEEE Conference on Open Systems*, 2016, pp. 130–135, doi: 10.1109/ICOS.2015.7377291 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84962803602&doi=10.1109%2fICOS.2015.7377291&partnerID=40&md5=3ab79ca67251106c8c4a0616856c05e7>
- [33] W. Zhang *et al.*, "OpenNetVM: A platform for high performance network service chains," in *HotMiddlebox 2016 - Proceedings of the 2016 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization, Part of SIGCOMM 2016*, 2016, pp. 26–31, doi: 10.1145/2940147.2940155 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85030151023&doi=10.1145%2f2940147.2940155&partnerID=40&md5=5c3aeb6cb406b34a1a37f4855559183e>
- [34] A. M. Joy, "Performance comparison between Linux containers and virtual machines," in *Conference Proceeding - 2015 International Conference on Advances in Computer Engineering and Applications, ICACEA 2015*, 2015, pp. 342–346, doi: 10.1109/ICACEA.2015.7164727 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84947976761&doi=10.1109%2fICACEA.2015.7164727&partnerID=40&md5=36cf0e250f42c1f17daced1154bb07ba>
- [35] A. Stanciu, "Blockchain Based Distributed Control System for Edge Computing," in *Proceedings - 2017 21st International Conference on Control Systems and Computer, CSCS 2017*, 2017, pp. 667–671, doi: 10.1109/CSCS.2017.102 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85027499223&doi=10.1109%2fCSCS.2017.102&partnerID=40&md5=12f6a51c5cad47558bce7f986df7d69a>
- [36] D. Taibi, V. Lenarduzzi, and C. Pahl, "Processes, Motivations, and Issues for Migrating to Microservices Architectures: An Empirical Investigation," *IEEE Cloud Computing*, vol. 4, no. 5, pp. 22–32, 2017, doi: 10.1109/MCC.2017.4250931. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85038613047&doi=10.1109%2fMCC.2017.4250931&partnerID=40&md5=b4889bdf1ae46384a04467f931220c0>
- [37] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-based electronic healthcare record system for healthcare 4.0 applications," *Journal of Information Security and Applications*, vol. 50, 2020, doi: 10.1016/j.jisa.2019.102407. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85075253060&doi=10.1016%2fj.jisa.2019.102407&partnerID=40&md5=4dbe2e6b8bf7df394060258fc5b70129>
- [38] M. T. Chung, N. Quang-Hung, M.-T. Nguyen, and N. Thoai, "Using Docker in high performance computing applications," in *2016 IEEE 6th International Conference on Communications and Electronics, IEEE ICCE 2016*, 2016, pp. 52–57, doi:

10.1109/CCE.2016.7562612 [Online]. Available:
<https://www.scopus.com/inward/record.uri?eid=2-s2.0-84988835849&doi=10.1109%2fCCE.2016.7562612&partnerID=40&md5=d57d2da885f2cf80d819593abf0118d3>

- [39] A. J. Younge, K. Pedretti, R. E. Grant, and R. Brightwell, "A Tale of Two Systems: Using Containers to Deploy HPC Applications on Supercomputers and Clouds," in *Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom*, 2017, vol. 2017-December, pp. 74–81, doi: 10.1109/CloudCom.2017.40 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85044282707&doi=10.1109%2fCloudCom.2017.40&partnerID=40&md5=7bcf63c2427c380d87f2c16b2aec69e3>
- [40] C. Ruiz, E. Jeanvoine, and L. Nussbaum, "Performance evaluation of containers for HPC," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9523, pp. 813–824, 2015, doi: 10.1007/978-3-319-27308-2_65. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84951948191&doi=10.1007%2f978-3-319-27308-2_65&partnerID=40&md5=158c3c0f9e1fb3dabc04483040f96603
- [41] J. Sparks, "Enabling Docker for HPC," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 16, 2019, doi: 10.1002/cpe.5018. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85057995453&doi=10.1002%2fcpe.5018&partnerID=40&md5=2287f5171d63784b55bfe4b97d3859b5>
- [42] A. Azab, "Enabling docker containers for high-performance and many-task computing," in *Proceedings - 2017 IEEE International Conference on Cloud Engineering, IC2E 2017*, 2017, pp. 279–285, doi: 10.1109/IC2E.2017.52 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85020179264&doi=10.1109%2fIC2E.2017.52&partnerID=40&md5=4fb9b431ca1003a969e3a80547729dbe>
- [43] C. K. Dehury and P. K. Sahoo, "Design and implementation of a novel service management framework for IoT devices in cloud," *Journal of Systems and Software*, vol. 119, pp. 149–161, 2016, doi: 10.1016/j.jss.2016.06.059. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84977271482&doi=10.1016%2fj.jss.2016.06.059&partnerID=40&md5=2beb768677d99e3505c2c23d24e024ed>
- [44] Q. Xu, C. Jin, M. F. B. M. Rasid, B. Veeravalli, and K. M. M. Aung, "Blockchain-based decentralized content trust for docker images," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 18223–18248, 2018, doi: 10.1007/s11042-017-5224-6. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85030856326&doi=10.1007%2fs11042-017-5224-6&partnerID=40&md5=9a4fa055a2f6e05c3a745ec14846fcb2>

- [45] F. Ramalho and A. Neto, "Virtualization at the network edge: A performance comparison," in *WoWMoM 2016 - 17th International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2016, doi: 10.1109/WoWMoM.2016.7523584 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84983806009&doi=10.1109%2fWoWMoM.2016.7523584&partnerID=40&md5=c3a768979d0f88b01582edb3f873c91f>
- [46] P. Mendki, "Docker container based analytics at IoT edge Video analytics usecase," in *Proceedings - 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages, IoT-SIU 2018*, 2018, doi: 10.1109/IoT-SIU.2018.8519852 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85057733275&doi=10.1109%2fIoT-SIU.2018.8519852&partnerID=40&md5=5f9a00d58cb858a6a71530a9cab5104a>
- [47] H. Sami and A. Mourad, "Dynamic On-Demand Fog Formation Offering On-the-Fly IoT Service Deployment," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 1026–1039, 2020, doi: 10.1109/TNSM.2019.2963643. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85083704763&doi=10.1109%2fTNSM.2019.2963643&partnerID=40&md5=f1aab3b690eb7f6ad4ff615609d902bd>
- [48] G. D. Putra, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "Trust Management in Decentralized IoT Access Control System," in *IEEE International Conference on Blockchain and Cryptocurrency, ICBC 2020*, 2020, doi: 10.1109/ICBC48266.2020.9169481 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85091518744&doi=10.1109%2fICBC48266.2020.9169481&partnerID=40&md5=cb6831715b02d3141a2769f578c12532>
- [49] T. Pfandzelter and D. Bermbach, "TinyFaaS: A Lightweight FaaS Platform for Edge Environments," in *Proceedings - 2020 IEEE International Conference on Fog Computing, ICFC 2020*, 2020, pp. 17–24, doi: 10.1109/ICFC49376.2020.00011 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85086479146&doi=10.1109%2fICFC49376.2020.00011&partnerID=40&md5=26ce765c34509e71c23b230f9b9fbd2a>
- [50] T. Goethals, M. Sebrechts, A. Atrey, B. Volckaert, and F. de Turck, "Unikernels vs containers: An in-depth benchmarking study in the context of microservice applications," in *Proceedings - 8th IEEE International Symposium on Cloud and Services Computing, SC2 2018*, 2018, pp. 1–8, doi: 10.1109/SC2.2018.00008 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85060246102&doi=10.1109%2fSC2.2018.00008&partnerID=40&md5=2e7e86dcb8bd0d9d1af4ff1a351ad85e>
- [51] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, and P. Merle, "Autonomic Vertical Elasticity of Docker Containers with ELASTICDOCKER," in *IEEE International Conference on Cloud Computing, CLOUD*, 2017, vol. 2017-June, pp. 472–479, doi: 10.1109/CLOUD.2017.67 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0->

85032204470&doi=10.1109%2fCLOUD.2017.67&partnerID=40&md5=8457fe7a95e
e1d2725403983b738d448

- [52] R. Morabito, "A performance evaluation of container technologies on Internet of Things devices," in *Proceedings - IEEE INFOCOM*, 2016, vol. 2016-September, pp. 999–1000, doi: 10.1109/INFOCOMW.2016.7562228 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84988892955&doi=10.1109%2fINFOCOMW.2016.7562228&partnerID=40&md5=ed800257e65930d74b0ad9e237005ace>
- [53] R. Morabito, "Virtualization on internet of things edge devices with container technologies: A performance evaluation," *IEEE Access*, vol. 5, pp. 8835–8850, 2017, doi: 10.1109/ACCESS.2017.2704444. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85023780909&doi=10.1109%2fACCESS.2017.2704444&partnerID=40&md5=6b4da7fb63e30c49eb80d4b6a6b0d2a3>
- [54] A. Ahmed and G. Pierre, "Docker-Pi: Docker container deployment in fog computing infrastructures," *International Journal of Cloud Computing*, vol. 9, no. 1, pp. 6–27, 2020, doi: 10.1504/IJCC.2020.105885. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85082185472&doi=10.1504%2fIJCC.2020.105885&partnerID=40&md5=9c75a0e654bff3ed4fc0c6fd0c587b46>
- [55] A. Ahmed and G. Pierre, "Docker container deployment in fog computing infrastructures," in *Proceedings - 2018 IEEE International Conference on Edge Computing, EDGE 2018 - Part of the 2018 IEEE World Congress on Services*, 2018, pp. 1–8, doi: 10.1109/EDGE.2018.00008 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85055629025&doi=10.1109%2fEDGE.2018.00008&partnerID=40&md5=cfd3e39e892642036f6f97c69317feb>
- [56] C. Anderson, "Docker," *IEEE Software*, vol. 32, no. 3, pp. 102–105, 2015, doi: 10.1109/MS.2015.62. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84928741645&doi=10.1109%2fMS.2015.62&partnerID=40&md5=1dbb01b253906fc7d7cffeab255d069d>
- [57] R. Peinl, F. Holzschuher, and F. Pfitzer, "Docker Cluster Management for the Cloud - Survey Results and Own Solution," *Journal of Grid Computing*, vol. 14, no. 2, pp. 265–282, 2016, doi: 10.1007/s10723-016-9366-y. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84963690007&doi=10.1007%2fs10723-016-9366-y&partnerID=40&md5=549dc0921ea43cbb44ed1f2bfe156e16>
- [58] T. Ueda, T. Nakaike, and M. Ohara, "Workload characterization for microservices," in *Proceedings of the 2016 IEEE International Symposium on Workload Characterization, IISWC 2016*, 2016, pp. 85–94, doi: 10.1109/IISWC.2016.7581269 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0->

84994792149&doi=10.1109%2fIIISWC.2016.7581269&partnerID=40&md5=4d5176224c8f26bb71dad3e667e46e66

- [59] S. Taherizadeh, V. Stankovski, and M. Grobelnik, "A capillary computing architecture for dynamic internet of things: Orchestration of microservices from edge devices to fog and cloud providers," *Sensors (Switzerland)*, vol. 18, no. 9, 2018, doi: 10.3390/s18092938. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85053075552&doi=10.3390%2fs18092938&partnerID=40&md5=2849d743089f70cbb46f1b0644f62c65>
- [60] C. Pahl, "Containerization and the PaaS Cloud," *IEEE Cloud Computing*, vol. 2, no. 3, pp. 24–31, 2015, doi: 10.1109/MCC.2015.51. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84937678684&doi=10.1109%2fMCC.2015.51&partnerID=40&md5=d945e8d09d4bc50ec74e2c54e52c0c43>
- [61] C. Dupont, R. Giaffreda, and L. Capra, "Edge computing in IoT context: Horizontal and vertical Linux container migration," in *GloTS 2017 - Global Internet of Things Summit, Proceedings, 2017*, doi: 10.1109/GIOTS.2017.8016218 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85029314185&doi=10.1109%2fGIOTS.2017.8016218&partnerID=40&md5=c8a08bce5339072af8cf1bc5e507b597>
- [62] H. v Netto, L. C. Lung, M. Correia, A. F. Luiz, and L. M. de Souza, "State machine replication in containers managed by Kubernetes," *Journal of Systems Architecture*, vol. 73, pp. 53–59, 2017, doi: 10.1016/j.sysarc.2016.12.007. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85009278874&doi=10.1016%2fj.sysarc.2016.12.007&partnerID=40&md5=20f85d0f9595a740ad504bfe41830344>
- [63] C.-C. Chang, S.-R. Yang, E.-H. Yeh, P. Lin, and J.-Y. Jeng, "A Kubernetes-Based Monitoring Platform for Dynamic Cloud Resource Provisioning," in *2017 IEEE Global Communications Conference, GLOBECOM 2017 - Proceedings, 2017*, vol. 2018-January, pp. 1–6, doi: 10.1109/GLOCOM.2017.8254046 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85046372373&doi=10.1109%2fGLOCOM.2017.8254046&partnerID=40&md5=eccb2611e5a1df030d85f4a3a041d21a>
- [64] L. Abdollahi Vayghan, M. A. Saied, M. Toeroe, and F. Khendek, "Deploying Microservice Based Applications with Kubernetes: Experiments and Lessons Learned," in *IEEE International Conference on Cloud Computing, CLOUD, 2018*, vol. 2018-July, pp. 970–973, doi: 10.1109/CLOUD.2018.00148 [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85057488012&doi=10.1109%2fCLOUD.2018.00148&partnerID=40&md5=887f0d831b37008f1ba41dff4a7f72a>
- [65] J. Shah and D. Dubaria, "Building modern clouds: Using docker, kubernetes google cloud platform," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019, 2019*, pp. 184–189, doi:

10.1109/CCWC.2019.8666479 [Online]. Available:
<https://www.scopus.com/inward/record.uri?eid=2-s2.0-85063895862&doi=10.1109%2fCCWC.2019.8666479&partnerID=40&md5=5aa69b0a496f41a4d12b6adf310eb01f>