

DEVOPS EN LA BASE DE DATOS

Robles Flores, Anthony Richard (2016056192), Porlles Carrillo, Diego Armando (2015050948), Sandoval Blas, Jesus Enrique (2016054467), Quispe Mamani, José Luis (2015053235)

Tacna, Perú

Abstract

DevOps is a conceptual framework for reintegrating development and operations of Information Systems. We performed a Systematic Mapping Study to explore DevOps. 26 articles out of 139 were selected, studied and summarized. Based on this a concept table was constructed. We discovered that DevOps has not been adequately studied in scientific literature. There is relatively little research available on DevOps and the studies are often of low quality.

We also found that DevOps is supported by a culture of collaboration, automation, measurement, information sharing and web service usage. DevOps benefits IS development and operations performance. It also has positive effects on web service development and quality assurance performance. Finally, our mapping study suggests that more research is needed to quantify these effects.

1. Resumen

DevOps se le puede denominar como un marco conceptual que tiene como finalidad la reintegración del desarrollo y las operaciones de los sistemas de información. Tomando en cuenta en respecto a un estudio de mapeo sistemático para explorar DevOps, de los cuales fueron seleccionados 139 artículos y elegidos 26 artículos estudiados y resumidos. En base a estos estudios se tomó la sorpresa que DevOps no se ha estudiado adecuadamente en la literatura científica puesto que la información acerca de DevOps puede ser aún confusa para algunas empresas u organizaciones.

En este artículo podremos resaltar que DevOps está respaldado bajo ciertas culturas como : colaboración, automatización, medición, intercambio de información y uso de servicios web beneficiando al desarrollo de SI y el rendimiento de las operaciones causando a la vez efectos positivos en el desarrollo del servicio web y el rendimiento del aseguramiento de la calidad. En este artículo se definirá como esta herramienta podría afectar en el desarrollo y operaciones en una organización.

2. Introducción

El desarrollo ágil puede dividir en interacciones el desarrollo de un sistema de software para la finalidad de obtener incrementos de software en estado funcional. A pesar de los beneficios del desarrollo ágil, se pueden originar nuevos retos debido a las brechas que aparecen en el momento de integrar, el realizar pruebas y poder desplegar a producción cada incremento de lo desarrollado. Hay que tomar en cuenta el tiempo de entrega de cada incremento que puede agregarse es un factor importante puesto que puede originar ventajas de competitividad a las organizaciones, permitiendo obtener retroalimentación de los clientes y de esta manera poder realizar mejoras continuas de los productos de software de los productos de software. Realizar manualmente estos procesos de integración, pruebas y despliegue de incrementos de software puede ocasionar en la mayoría de los casos retrasos en la planificación de los proyectos y esto puede ser propenso a errores con un resultado no confiable.

DevOps es un conjunto de principios y prácticas que pueden optimizar el tiempo de entrega de software, gestionar la infraestructura como el código y poder mejorar la experiencia del usuario en base a la retroalimentación de sus comentarios. DevOps nos ofrecen prácticas las cuales son: i) la integración continua construye todo el software y ejecuta un conjunto de pruebas con cada incremento, ii) el despliegue continuo toma el software creado en la integración y lo despliega en un entorno de operaciones de una manera automática, y iii) la entrega continua proporciona la capacidad de liberar en el entorno de operaciones nuevas versiones de software varias veces al día.

DevOps es un método de reconfiguración dinámica e incremental de arquitecturas de servicios Cloud basado en el DSDM (desarrollo de software dirigido por modelos) , este método tiene como objetivo poder apoyar a los desa-

rolladores de software en la integración incremental de servicios en entorno Cloud, esto provee un conjunto de actividades que soportan i) el diseño de la integración, ii) la implementación de la integración, y iii) el despliegue y reconfiguración arquitectónica.

3. Marco Teórico

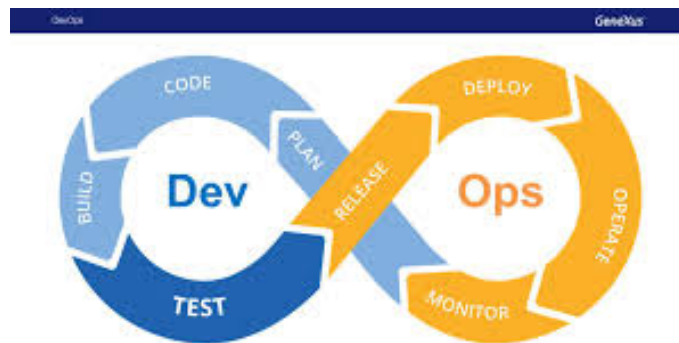
3.1. Definición

DevOps es una metodología de desarrollo de software que apunta a reunir equipos de desarrollo de software y operativos de tecnología de la información. Es un concepto que fomenta una cultura de colaboración entre estos dos equipos que históricamente trabajaron en sus propios silos separados, desde la fase de diseño inicial hasta el lanzamiento del producto.

DevOps es una metodología que combina el desarrollo de software (Dev) con las operaciones (Ops). La intención es permitir la comunicación entre los equipos para que puedan construir, probar y lanzar software más rápidamente y con mayor eficiencia y velocidad.

Al combinar estos dos equipos y procesos distintos, se promueve la integración continua, la implementación continua, las pruebas automatizadas y la transparencia en los repositorios de código.[5]

[?]

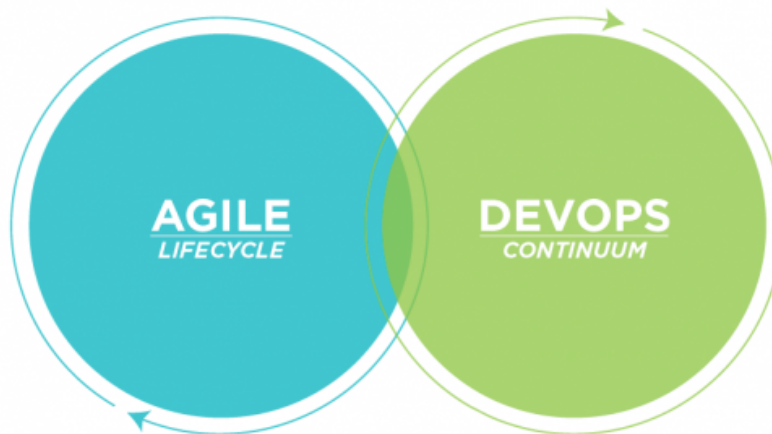


3.2. ¿Qué es Agile?

La metodología ágil es también una metodología de desarrollo de software que surgió alrededor del año 2001, cuando se presentó el manifiesto ágil. Emplea cuatro valores y doce principios que ayudan a construir una cultura de desarrollo de software "ágil".

En términos generales, ágil fomenta la adopción y una mentalidad de liderazgo que promueve el trabajo en equipo, la autoorganización y la responsabilidad. Más importante aún, el enfoque ágil se enfoca más en alinear continuamente el desarrollo con las necesidades y tendencias del cliente, incluso cuando esas necesidades y tendencias cambian al final del proceso de desarrollo.

Agile incorpora un conjunto de principios que ayudan a individuos, equipos y unidades más grandes a trabajar juntos. La "mentalidad ágil" se enfoca más en las personas que en los procesos y herramientas. Una organización ágil se adapta y aprende sobre el cambio constante que les permite identificar nuevas oportunidades y añadir más valor para los clientes. "SQL-92.º" "SQL2".[5]



3.3. Requisitos para implementar una cultura DevOps

Lo primero es comunicar que la cultura DevOps se va a implementar, mencionando los beneficios y haciendo que el equipo se sienta parte del cambio. Luego explicar las acciones que se van a tomar, no importa que no sean desarrolladores.

La comunicación y colaboración para tener una cultura DevOps son cruciales, es un trabajo entre los desarrolladores y los equipos encargados de la infraestructura de los servidores.

La tarea principal será la automatización, reducir los tiempos de despliegue del producto y mantener una calidad de desarrollo y estabilidad para el usuario. Debes tener claro que esto será un proceso sin fin, por lo que cada mejora te dará el tiempo que dedicarás para innovar el proceso y hacerlo cada vez mejor.[3]

3.4. Contradicciones en DevOps de base de datos

Como podemos ver, las bases de datos se ha vuelto una parte muy importante en el proceso de modernización de las aplicaciones, lo cual puede presentar un problema intratable.

En DevOps para entornos de prueba de bases de datos, las organizaciones deben generar rápidamente clones del sistema para ofrecer datos que reflejen con precisión el entorno de producción, al tiempo que protegen la información corporativa sensible. Por otro lado, los cambios de la base de datos en la producción representan el abismo más amplio entre las técnicas de desarrollo de aplicaciones ágiles y la capacidad de desplegarlas en infraestructuras de TI del mundo real.[7]

3.5. DevOps de base de datos desafía los problemas de pruebas

Normalmente, los desarrolladores son los primeros en descubrir las dificultades técnicas de trabajar con bases de datos en entornos de prueba automatizados. "No tienen los datos correctos", dijo Brandon Cipes, director gerente de DevOps en cPrime, una firma de consultoría ágil en Foster City, California. "¿Cómo hacen que el entorno de prueba ilustre la producción, y cómo mantienen ese ambiente actualizado y seguro?". Las herramientas de gestión de datos de prueba están disponibles de proveedores como DBmaestro y CA Technologies, pero es aquí donde empiezan a surgir desafíos de personas y procesos, dijo Nirmal Mehta, tecnólogo jefe de la firma de consultoría Booz Allen Hamilton.[7]

3.6. *Las bases de datos de producción presentan desafíos adicionales de DevOps*

Para abordar estos problemas, algunas organizaciones han movido las aplicaciones de producción a bases de datos NoSQL, que no requieren esquemas. Sin embargo, muchos han retrocedido después de que los sistemas NoSQL presentaran problemas de gestión en la producción, dijo Cook. "Muchos equipos se dieron cuenta de que, cuando la gente decía que MongoDB era más simple, significaba para la instalación", dijo Cook.[2]

"Pero para una aplicación de producción, la gente termina por tener que construir un esquema y un sistema de sanitización de datos encima de todos modos, y si no eres un DBA [administrador de bases de datos], NoSQL se convierte en un desastre muy fácilmente". "De repente, Postgres se convirtió en el mejor de ambos mundos", dijo Cook. Según Rob Steward, vicepresidente de desarrollo de producto de Progress Software, "una buena práctica de DevOps liberará a los desarrolladores para que se centren en hacer lo que mejor saben hacer: escribir software. DevOps elimina el trabajo y las preocupaciones de la puesta en producción del software una vez que está escrito".[3]

3.7. *¿Por qué se usa DevOps?*

DevOps permite a los equipos de desarrollo ágiles implementar la integración continua y la entrega continua. Esto les ayuda a lanzar productos más rápidamente en el mercado. Otras razones importantes son: Es Predecible: DevOps ofrece una tasa de fallas significativamente menor de las nuevas versiones.[4]

- 1. Reproducibilidad: Permite que la versión anterior pueda restaurarse en cualquier momento.
- 2. Mantenimiento: Proceso de recuperación sin esfuerzo en caso de que una nueva versión falle o deshabilite el sistema actual.
- 3. Tiempo de comercialización: DevOps reduce el tiempo de comercialización hasta un 50 por ciento mediante la entrega de software simplificada. Este es particularmente el caso de las aplicaciones digitales y móviles.
- 4. Mayor calidad: DevOps ayuda al equipo a mejorar la calidad del desarrollo de aplicaciones ya que incorpora problemas de infraestructura.

- 5. Riesgo reducido: DevOps incorpora aspectos de seguridad en el ciclo de vida de la entrega del software. Ayuda en la reducción de defectos en todo el ciclo de vida.
- 6. Resistente: el estado operativo del sistema de software es más estable, seguro y los cambios son consultables.
- 7. Eficiencia de costes: DevOps ofrece rentabilidad en el proceso de desarrollo de software, que siempre es una aspiración de la gestión de las empresas de TI.
- 8. Rompe una base de código más grande en pequeños trozos: DevOps se basa en el método de programación ágil. Por lo tanto, permite dividir las bases de códigos más grandes en trozos más pequeños y manejables.

3.8. *¿Cuándo no adoptar DevOps?*

No debe usarse en una aplicación de misión crítica como bancos, energía y otros sitios de datos confidenciales. Dichas aplicaciones necesitan estrictos controles de acceso en el entorno de producción, una política detallada de gestión de cambios, política de control de acceso a los centros de datos.

3.9. *Principios de DevOps*

Aquí, hay seis principios que son esenciales al adoptar DevOps:

- Acción centrada en el cliente: el equipo de DevOps debe tomar medidas centradas en el cliente para que constantemente inviertan en productos y servicios.
- Responsabilidad de extremo a extremo: el equipo de DevOps debe proporcionar soporte de rendimiento hasta que se ocurra el fin de vida. Esto mejora el nivel de responsabilidad y la calidad de los productos diseñados.
- Mejora Continua: la cultura DevOps se centra en la mejora continua para minimizar el desperdicio. Continuamente acelera la mejora del producto o los servicios ofrecidos.
- Automatiza todo: la automatización es un principio vital del proceso DevOps. Esto no es solo para el desarrollo de software, sino también para todo el panorama de la infraestructura.

- Trabajar como un solo equipo: en el rol de DevOps, el del diseñador, desarrollador y evaluador ya están definidos. Todo lo que tenían que hacer es trabajar como un equipo con total colaboración.
- Monitorea y pruebe todo: es muy importante que el equipo de DevOps tenga un sólido monitoreo y procedimientos de prueba.[1]

El enfoque de DevOps necesita cambios frecuentes e incrementales en las versiones de código, lo que significa frecuentes despliegues y pruebas. Aunque los ingenieros de DevOps necesitan codificar de vez en cuando desde cero, es importante que tengan las bases de los lenguajes de desarrollo de software. Un ingeniero de DevOps trabajará con el personal del equipo de desarrollo para abordar la codificación y secuencias de comandos necesarias para conectar elementos de código, como bibliotecas o kits de desarrollo de software.

4. Analisis

4.1. DevOps en entorno de pruebas de Bases de Datos

DevOps para el entorno de prueba de Base de Datos más que solución podrían ocasionar problemas en el entorno con referencia al uso de Contenedores y el tema del precio

Según el siguiente Párrafo.

.Extraer los datos de producción de una cola y clonar el tráfico de producción para obtener insumos realistas puede ocasionar costos de infraestructura que muchas organizaciones no quieren pagar”, dijo Nirmal Mehta, tecnólogo jefe de la firma de consultoría Booz Allen Hamilton.[6]

Los contenedores son el furor en las pruebas de aplicaciones automatizadas porque los desarrolladores pueden crearlos rápidamente y luego destruirlos, pero las aplicaciones de base de datos con estado aún no se prestan bien a la contenerización.

”Hasta cierto punto, hay una incongruencia fundamental entre las bases de datos y los contenedores”, dijo E.T. Cook, principal defensor de la firma de consultoría etc.io, que ayuda a las empresas a desplegar contenedores.

Si bien hay herramientas que te permiten el almacenamiento persistente en aplicaciones contenerizadas, pero los enfoques más probados no resuelven el problema de la verdadera portabilidad de datos, dijo Cook.[6]

Existen productos que ofrecen un almacenamiento persistente de contenedores verdaderamente distribuido, pero las empresas todavía deben aprender cómo administrar e integrar estos sistemas.

A pesar de la disponibilidad de estas herramientas, muchas empresas aún desarrollan sus propias capas de abstracción para administrar múltiples tipos de bases de datos. Estas capas de abstracción actúan como amortiguadores entre las aplicaciones y las bases de datos, para que los cambios en cualquiera de ellos no interrumpan al otro, dijo Cipes de cPrime.[6]

”Las bases de datos de producción presentan desafíos adicionales de DevOps”

Incluso cuando los problemas de DevOps en bases de datos se resuelven en la prueba, el abismo sigue siendo amplio entre el entorno de prueba y la infraestructura de producción donde se ejecutarán realmente las aplicaciones. Las bases de datos de producción pueden ser engorrosas para actualizar, requieren el conocimiento tribal de esquemas de datos complejos para acceder y operar, y son lentas para escalar. Pero también las Bases de datos Avanzan con el pasar del tiempo incorporando objetos no esquemáticos, que permiten que los datos no estructurados sean parte de las consultas contra índices y con el tiempo de seguro los problemas que existían anteriormente cada vez se volverá tan pequeño que el uso de devOps ya no será un impedimento para que las empresas decidan usarlo , siendo uno de las metodologías mas favoritas que hay en la actualidad ofreciendo Resultado Tangibles y adoptado por las organizaciones un Total de 1425 según la información en 2014. Devops: Salir victoriosos en la economía de las aplicaciones.[8]

5. Conclusiones

5.1. Conclusión

En conclusión, sacaremos una definición simple de DevOps con la que podamos estar de acuerdo, y este concepto es que DevOps es una metodología de desarrollo de software basada en la integración entre desarrolladores de software basada en la integración entre desarrolladores y administradores de sistemas, que permite que los desarrolladores puedan enfocarse solo en desarrollar y puedan desplegar su código en segundos.

Referencias

- [1] Cristina, J. R. (nea). El legendario origen del movimiento devops. Recuperado de <http://cort.as/-P7lp>. Accedido 31-08-2019.
- [2] Cristina, J. R. (neb). Qué es devops 2015. Recuperado de <http://cort.as/-P7oQ> . Accedido 28-08-2019.
- [3] Daneva, M. (ne). Devops literature review. Recuperado de <http://cort.as/-P7Ff>. Accedido 18-08-2019.
- [4] IntelDig (ne). ¿por qué se usa devops?,¿cuándo no adoptar devops?,principios de devops. Recuperado de <http://cort.as/-P7Ne>. Accedido 31-08-2019.
- [5] It, U. (ne). Agile and devops. Recuperado de <http://cort.as/-P7I7>. Accedido 18-08-2019.
- [6] Pariseau, B. (2017). Contradicciones en devops de base de datos en-mañá la automatización de ti. Recuperado de <https://bit.ly/2jVHzk9>. Accedido 30-08-2019.
- [7] Pariseau, B. (ne). Guía esencial: Las bases de datos dan soporte a las tendencias de ti 2017. Recuperado de <http://cort.as/-P7o9> . Accedido 28-08-2019.
- [8] Technologies, C. (2014). Devops: Salir victoriosos en la economía de las aplicaciones. Recuperado de <https://bit.ly/2lP06Px>. Accedido 30-08-2019.