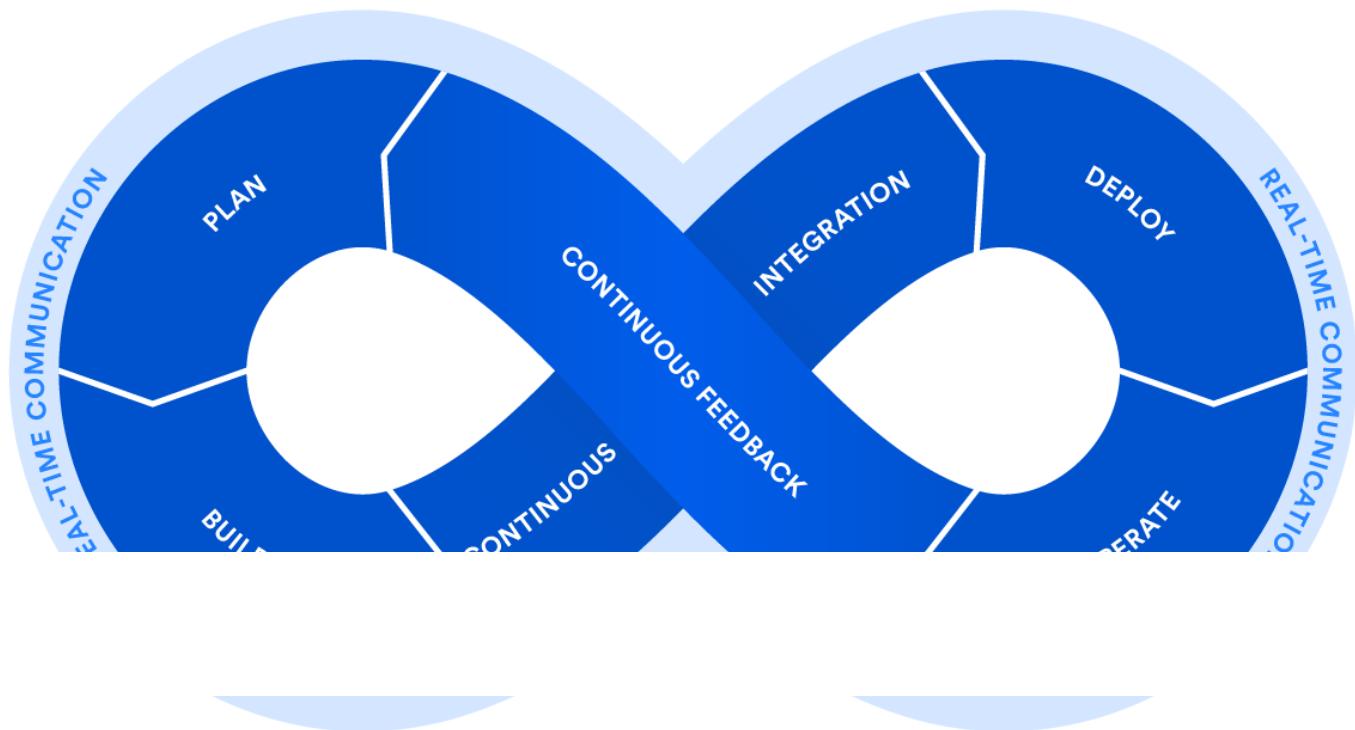


# El ciclo de DevOps, una guía para iniciarse en las fases que lo componen

Compartir



## 5 COMENTARIOS

1 Abril 2018 - Actualizado 13 Abril 2018, 12:04 **JUAN QUIJANO**

Hay términos que están de moda, tienen mucha capacidad de empoderamiento, y producen movimientos innovadores en las empresas. Sobre todo, las dedicadas a negocios relacionados con la economía digital.

Uno de ellos es, junto con microservicios, Agile, o transformación digital, **el término DevOps**.

Este es un concepto, quasi una filosofía, del cual en este artículo **voy a centrarme en la descripción de**

SUSCRÍBETE A GENBETA DEV

Recibe un email al día con nuestros artículos:

Tu correo electrónico  SUSCRIBIR

Síguenos



PUBLICIDAD

**las fases del ciclo iterativo que lo componen;** con el objetivo de aclarar conceptos básicos que debo tener

Compartir

f 219



## Fases en DevOps



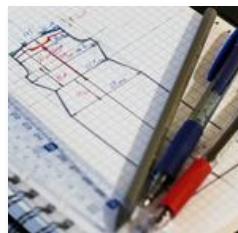
En la actualidad, DevOps se puede definir como **un símbolo de infinito o un círculo que define las diferentes áreas y fases** que lo componen:

- Plan
- Desarrollo
- Integración continua

- Monitorización

TE RECOMENDAMOS

Es importante comprender que es una de las múltiples representaciones, no el canon definitivo. Haciendo simplificaciones totalmente válidas en forma de cuatro fases principales, o descomposiciones detalladas de cada una de ellas.



3 patrones de diseño imprescindibles que deberías conocer para tu sistema en cloud: Retry, Valet Key y Sharding

Otra idea imprescindible de interiorizar es que se trata de la definición de un flujo iterativo, por lo cual diferentes procesos pueden estar comprendidos en diferentes fases de forma orgánica y superpuesta, siempre ajustándose a los conceptos fundamentales de valor y mejora continua.



Nueve programas de detección de movimiento para convertir tu PC en un sistema de videovigilancia

**Debo evitar caer en la tentación de considerarlo como un ciclo en cascada,** en donde las fases están

delimitadas rígidamente por una frontera que las separa, o que solamente se puede iniciar una fase



Experiencias en el desarrollo de

Compartir

f 219



Ahora, voy a mirar con mas detalle cada tase, permitiéndome una licencia muy habitual en los procesos DevOps, que es utilizar el framework Scrum (y sus arquetipos) como metodología de trabajo para hacer más sencillas las explicaciones.



PUBLICIDAD

## Gestión y planificación



**conjunto mínimo de funcionalidades que permitan aportar Valor en cada iteración**, los criterios de aceptación a cumplir y la definición de acabado; para cada una de las fases y en el conjunto del proyecto.

Esto se constituye como una Pila de Producto viva, que está soportando continuadamente un proceso de "jardinería", desde un punto de vista de negocio, la cual alimenta a las diferentes fases de desarrollo y operaciones; y que aborda los cambios y evoluciones según un proceso de mejora continua basado en un feedback temprano y continuado.

Para ello utilizare las "*liturgias*" de Scrum, como son las reuniones de Planificación de la iteración y la Revisión de la iteración; pero sin, por ello, dejar de **tener una comunicación e implicación constante entre negocio y el equipo técnico**. Siendo imprescindible que Negocio y Gestión se formen en las herramientas y métricas diseñadas para que tengan una visibilidad veraz y suficiente del desarrollo del proyecto.

## Desarrollo, construyendo código

Compartir

f 219



Esta fase es en donde se construye. Sea picando código, diseñando infraestructura,

automatización de las acciones repetitivas o complejas, y que debería ser uno de los primeros peldaños que escalar para implantar DevOps en una organización.

Si tuviera que **resumir en una sola palabra el concepto más importante de esta fase, esta sería “pruebas”**.

Ya sea en una aplicación de gestión, operaciones con datos o el despliegue de infraestructura virtual; siempre voy a trabajar en código – ya sea con un lenguaje de programación o de scripting; el cual debe ser almacenado en un gestor de código que me permita operaciones básicas como históricos, ramas, versionado, etc.

Pero con esto no es suficiente y **cada pieza construida debe incluir (obligatoriamente) sus propias pruebas automatizadas**. Es decir, los mecanismos con los que el propio sistema pueda asegurarse de que lo que hemos realizado es correcto, no falla, no hace fallar a otras partes, cumple los criterios de aceptación, y señala de forma temprana los errores que surgen en todo desarrollo.

De hecho, este es un camino imperativo para adoptar Devops, desde sus más incipientes estadios.

Compartir

f 219



rollback; luego empezar a incluir las pruebas automatizadas, lo cual va a producir una transformación profunda en las técnicas de codificación (desacoplamiento, segregación, modularización, etc); por último, se llega a la orientación de lo construido hacia las fases siguientes, incluyendo la transformación del propio flujo de trabajo.

## Integración continua, o cómo dormir tranquilo



Aunque en esta fase y la anterior la mayoría de los autores nos centramos en un punto de vista de desarrollo, realmente **la llegada de DevOps y los conceptos de Infraestructura como código, hacen que IT también sea pleno participante de esta fase.**

La integración continua es automatizar el mecanismo de revisión, validación, prueba y alertas del valor construido en las iteraciones, desde un punto de vista global.

Es decir, mi singular funcionalidad o característica, que he construido en mi entorno de desarrollo, junto con las pruebas automáticas que aseguran su correcto funcionamiento, **son**

Compartir

f 219



Así, lanzando todas las pruebas que incluye cada funcionalidad, mas las pruebas de integración de toda la aplicación, más las pruebas funcionales, más las pruebas de aceptación, más los análisis de calidad del código, más las pruebas de regresión, podré estar seguro de que mi aplicación sigue funcionando correctamente.

Y si algo falla, saltará la alerta temprana indicando en qué pieza y en qué línea está rompiendo mi sistema.

Así que, **cuanto más me acerque al momento de iniciar el camino crítico del despliegue, más tranquilo estaré** porque más pruebas incluyen mi trabajo.

## Despliegue automatizado



Compartir

f 219



Desplegar, en las organizaciones clásicas, *siempre ha sido un dolor*. Dos roles (Dev e IT) con objetivos e intereses divergentes se encuentran en una batalla de incomunicación y recelo mutuo para publicar la aplicación en los diferentes entornos de trabajo: desarrollo, integración, calidad/test, preproducción, producción, etc.

Como en toda cadena, es fácil romper por el eslabón más débil, y **cuantos más pasos existan en los procesos de despliegue, más posibilidades de fallo humano se suman**.

Así, DevOps promueve la automatización de los despliegues por medio de herramientas y scripts, con el objetivo último de que todo el proceso se resuelva con un botón de aprobación o, idealmente, la activación de una característica.

Entre cada entorno de despliegue, hay que tener muy en cuenta la administración del contexto (crear, configurar y destruir entornos); realizar y superar las pruebas específicas de cada uno (como pueden ser pruebas de rendimiento, resiliencia, funcionales, de seguridad o

de UX); y administrar la gestión de la configuración (CMDB) de acuerdo con las complejas necesidades de los diferentes contextos de despliegue.

Compartir

f 219



llegada del concepto Cloud con sus capacidades de infraestructura como código, que fuerza un cambio en el paradigma de la gestión de la infraestructura. **Que pasa de una gestión de recursos finitos a una gestión basada en una optimización permanente de costes.**

## Operaciones, velando por el buen funcionamiento



Es una minoría las aplicaciones que son puestas en producción y no requieren de un trabajo constante en su optimización, evolución, o soporte. Pero, además, debo tener en cuenta todas las operaciones relacionadas con su funcionamiento que deben realizarse de forma continuada durante toda la vida del software.

Así tendrá, **el ajuste de los recursos de acuerdo con la demanda o las características de crecimiento de las aplicaciones;** la modificación dinámica de la infraestructura por causas de seguridad, rendimiento, disponibilidad, etc.; o la optimización de procesos y procedimientos que requieren cambios en el contexto de ejecución y explotación.

En esta fase, aplicará como anillo al dedo la adopción del concepto de Nube – sea pública, privada o hibrida- en dónde las operaciones puedan explotar las capacidades de

escalabilidad, persistencia, disponibilidad, transformación, resiliencia y seguridad que ofrecen este tipo de plataformas.

Compartir

f 219



**operaciones**, de forma que vuelvo a mitigar los tallos a causa de error humano.

## Monitorización, o el arte de medir



Esta última fase de un proceso DevOps, **es una fase permanente y que se aplica a todo el**

salud de las aplicaciones y su infraestructura, siendo esto el histórico de las mediciones durante un periodo de tiempo, que me muestran la evolución del sistema.

Tiene una vertiente reactiva, en donde de acuerdo con los resultados iré ajustando o modificando la plataforma; y otra proactiva en la cual un proceso de aprendizaje continuo me va a permitir adelantarme a las necesidades y riesgos.

Lo que no se define no se puede medir. Lo que no se mide, no se puede mejorar. Lo que no se mejora, se degrada siempre. William Thomson.

Menú no todo es tecnología, y en el  
nbitos y niveles del ciclo Dev  
La fase de Plan, o de forma inmedi

r el feedback continuo de todos en la siguiente iteración durante  
tales.



Monitorizaré, analizaré y mediré todo aquello que me pueda aportar una visión general del estado actual del proyecto (en su definición más amplia), incluyendo todas las dependencias

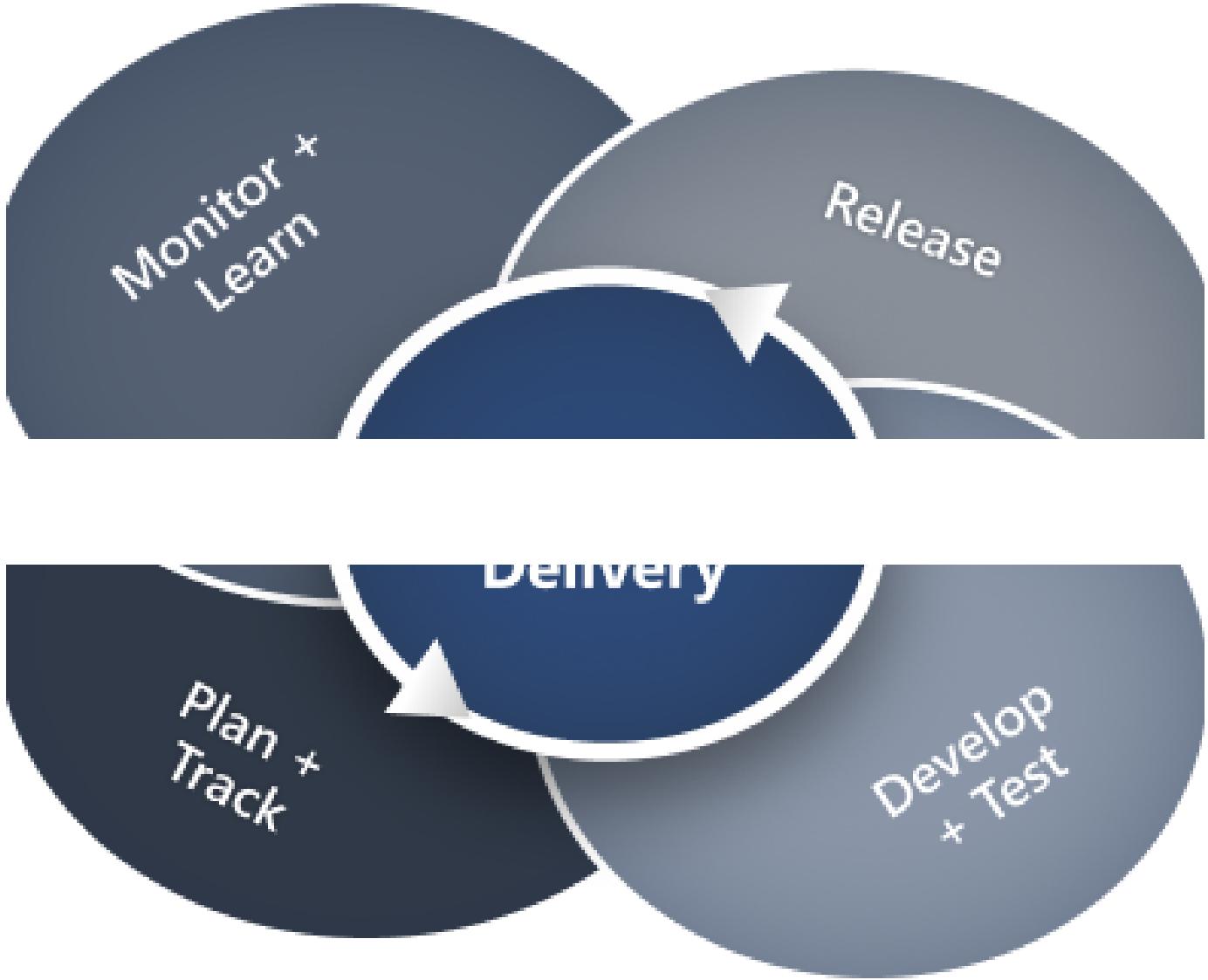
Compartir

f 219



**Y por medio de la realización de retrospectivas, completaré el proceso de Kaizen (mejora continua) del proyecto**, incluyendo todos los orígenes relacionados con el desarrollo positivo de los trabajos.

## Sobre las fases del ciclo DevOps



He mostrado una visión generalizada e idílica de un ciclo DevOps completo, siguiendo una estructura lógica basada en la experiencia y en la profusa literatura que hay publicada.

**La realidad es bastante más compleja tanto en la implantación como en la ejecución;** pero sin duda el mayor escollo son los problemas de comunicación entre los equipos con

objetivos e intereses diferentes, y la desconfianza de salir de mi zona de confort siguiendo una "moda".

Compartir

f 219



productividad, del Time to Market, o de la flexibilidad de la empresa frente a las demandas del negocio y los clientes; **si no también en los factores humanos, de mejora de la calidad de vida, que hay que valorar en su justa y positiva medida.**

Compartir

f 219



Temas:

PROGRAMACIÓN EN LA NUBE

DEVOPS

MICROSERVICIOS

PUBLICIDAD

## TAMBIÉN TE PUEDE GUSTAR



Porqué, para qué, cuándo y cómo migrar al Cloud computing



Qué es blockchain: la explicación definitiva para la tecnología más de moda



Alcanza la gloria y llévate 10.000€ en premios con el Mutua

Compartir

f 219



Así funcionará la red de internet mundial de más de 4.000 satélites que Elon Musk comienza a construir este sábado



Crea experiencias de realidad aumentada para Snapchat con la aplicación Lens Studio, disponible para PC y Mac



Las cuatro fases de Ethereum: qué es y qué supone Metropolis para esta criptomonedada



Esta plataforma de cursos te pagará criptomonedas por aprender programación y otras disciplinas tecnológicas



¿Qué pruebas debemos hacerle a nuestro software y para qué?

## 5 comentarios

OPCIONES

ESCRIBIR COMENTARIO X



Entra para comentar

Normas

## Compartir

f 219



- Recibir por e-mail:

- Todos
  - Sólo respuestas a los míos
  - Nada

PUBLICAR



dani77

4 Abr., 10:15 #1

Una vez más leo un artículo sobre DevOps y me quedo como estaba. Se hecha de menos algo más que teoría.

Más o menos viene a ser que "los de sistemas tienen que saber cosas de programación

"Hay que tener bien definido lo que hay que hacer"

"Hay que tener desacopladas las cosas"

"Hay que probar las cosas"

"Hay que automatizar las pruebas"

"Hay que detectar los errores cuanto antes"

"Hay que hacer que el despliegue sea automático o casi"

Quien haya tenido la gran suerte de participar en un proyecto en que se funcione así, enhorabuena, seguro que ha tenido maravillosos resultados, yo de momento no (pero no pierdo la esperanza!)



CERRAR RESPUESTAS X



## Respondiendo a dani77

**juanquijano**

"2

Compartir

f 219



Tienes toda la razón. De hecho, yo voy dos cursos (como trainer certificado) diferentes de DevOps, el profesional y el master. Ambos de unas 20 horas (que yo alargaría sin duda a las 25), y que se quedan cortos por todo el material que hay que transmitir.

Devops es un mundo enorme que, además, he obviado su acercamiento desde Lean.

Pero viendo la respuesta al artículo, creo que voy a seguir hablando de ello e ir profundizando sobre técnicas de implantación. Incluso llegando a las herramientas.

Muchas gracias por tu opinión.

**diegoparedes**#2  
5 Abr., 8:36

interesante articulo, muchas gracias por el aporte

  
Respondiendo a  
**diegoparedes****CERRAR RESPUESTAS** X**juanquijano**#4  
5 Abr., 12:28

Muchas gracias a tí, por leerte el artículo. Ojalá te sea útil o inspirador.

**LINUBE**

LINUBE

#5  
13 Abr., 12:04

Gracias por el aporte Juan

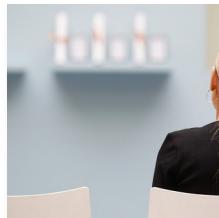
[Compartir](#)

f 219



PUBLICIDAD

## RECOMENDADO EN MAGNET



El problema del máster de Cifuentes no es simbólico. Una renuncia simbólica no lo arreglará



En estos años los idiomas de las dos Coreas han cambiado tanto que sus ciudadanos a veces no se entienden



¿Eres un "late sleeper"? Malas noticias: echarte a dormir tan tarde te está matando

## Compartir

f 219



**RECIBE UN EMAIL AL DÍA CON LOS ARTÍCULOS DE GENBETA DEV:**

**Tu correo electrónico**

Síguenos    

# EN GENBETA DEV HABLAMOS DE...

## Formación

## Herramientas

## Curiosidades

Open Source

## Frameworks

## Herramientas de desarrollo

## Eventos para Desarrolladores

java

[VER MÁS TEMAS](#)

## Buscar en Genbeta Dev



**SUBIR** ▲

[Xataka](#)[Tendencias](#)[Motorpasión](#)[El Blog Salmón](#)[Espinof](#)[Xataka Móvil](#)[Tendencias Belleza](#)[Motorpasión Moto](#)[Pymes y Autónomos](#)[Diario del Viajero](#)[Compartir](#)[f 219](#)[Xataka Smart Home](#)[Bebés y Más](#)[Xataka Windows](#)[Vitónica](#)[Xataka Ciencia](#)[Decoesfera](#)[Applesfera](#)[Vida Extra](#)[Genbeta](#)[Genbeta Dev](#)[Magnet](#)[Compradiccion](#)[Xataka eSports](#)

## LATINOAMÉRICA

[Xataka México](#)[Directo Al Paladar](#)[Motorpasión México](#)

## PARTICIPAMOS EN

[Nobbot](#)[Mi Mundo Philips](#)[Circula Seguro](#)[En Naranja](#)[Bluemagazine](#)[Tecnología de tú a tú](#)[Coca-Cola Journey  
España](#)[Circula Seguro PT](#)[Sage Experience](#)[Blog Lenovo](#)[Coca-Cola Journey  
México](#)[Corriente Eléctrica](#)[Bloggin Zenith](#)[eSports Unlocked by  
Orange](#)[Coca-Cola Journey  
Portugal](#)[Seguros de tú a tú](#)[Inget by acer](#)[Zona Coca-Cola](#)[Cervezas Alhambra](#)[Mahou Rentabilibar](#)