

Continuous Deployment



- ¿Que es?
- Beneficios
- Ejemplo de steps en gitlab
- Continuous delivery vs Continuous deployment

¿Que es?

Continuous Deployment es una estrategia en el desarrollo de software en la que los cambios de código de una aplicación se publican automáticamente en el entorno de producción. Esta automatización está impulsada por una serie de pruebas/steps predefinidas. Una vez que las nuevas actualizaciones pasan esas pruebas, el sistema envía las actualizaciones directamente a los usuarios del software



Beneficios



Beneficios

- **Entrega de software acelerada**

La ventaja más importante del desarrollo continuo es que acelera la ejecución de cambios de código al eliminar los cuellos de botella en cada etapa del proceso. Permite a los usuarios finales beneficiarse de las nuevas funciones más rápidamente porque no hay un tiempo de espera prolongado de semanas o meses

- **Comentarios más rápidos**

Un desarrollador solo necesita esperar unos minutos para monitorear el impacto de los nuevos cambios.. Cuando el comportamiento del cliente indica la necesidad de un giro rápido o un cambio de enfoque, la capacidad de publicar actualizaciones rápidamente es beneficiosa

- **Reducción de riesgos de desarrollo**

Cuando los equipos se integran de manera consistente y frecuente, reducen considerablemente los riesgos potenciales, ya que siempre conocen el estado actual del sistema

- **Reducción de procesos manuales**

La implementación continua alienta a los ingenieros a automatizar la mayor cantidad de procesos de desarrollo de software y lanzamiento de código, incluidas las pruebas de lanzamiento. La automatización permite a los desarrolladores impulsar nuevas versiones más rápidamente y ahorrar el tiempo que normalmente se dedica a los procesos manuales

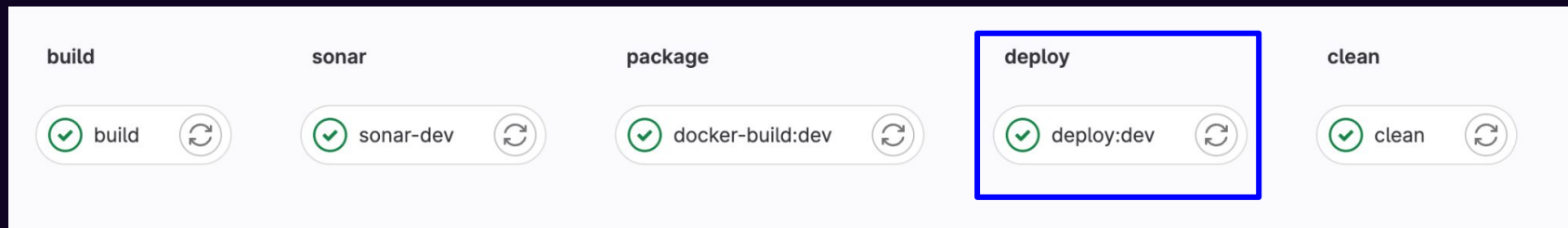


Ejemplo de steps en gitlab



Ejemplo de steps en gitlab

Diversos steps que se ejecutan al realizar un merge dentro una rama específica (steps de muestra)



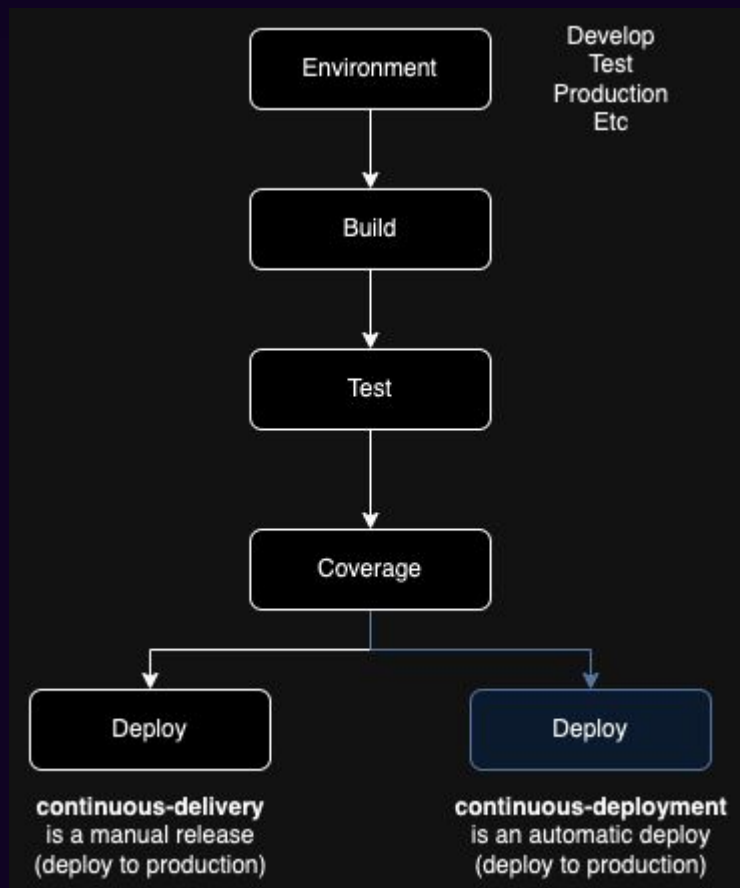
Deploy automatico

Continuous delivery vs Continuous deployment



Continuous delivery vs Continuous deployment

La principal diferencia entre **Continuous delivery** y **Continuous deployment** radica en la hora de desplegar los cambios. En Continuous delivery se debe realizar de manera manual el deploy, lo cual permite un último control pero también genera dependencia humana. Por otra parte tenemos a Continuous deployment el cual realiza todos los procesos de manera automática



Referencias

- **Ibm:**
<https://www.ibm.com/topics/continuous-deployment#:~:text=Continuous%20deployment%20is%20a%20strategy,directly%20to%20the%20software's%20users>
- **Atlassian:**
<https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment>

