

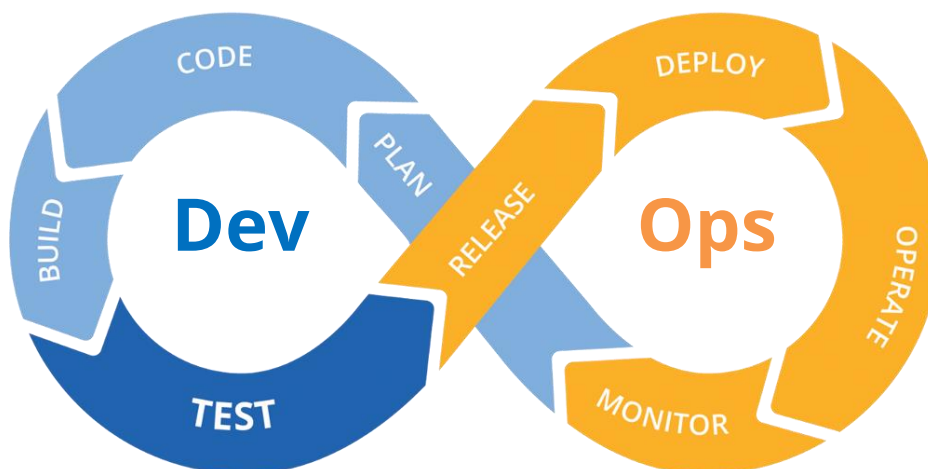
DevOps con GeneXus

Introducción



DevOps es una metodología que integra la creación de una aplicación, con las operaciones posteriores al desarrollo, como la puesta en producción, el mantenimiento y la evolución del software, en un ciclo sin fin. Veamos como se aplica esto en GeneXus.

Definición de DevOps



DevOps es un conjunto de prácticas ágiles que combina el desarrollo de software “Dev” y las operaciones de tecnología de la información “Ops” para acortar el ciclo de vida del desarrollo de sistemas y ofrecer software de forma continua.

El ciclo “Dev” (desarrollo) incluye el diseño, desarrollo, codificación, construcción y test de la aplicación. El Ops (Operación) incluye la liberación, puesta en producción, operación y monitoreo de la aplicación. Este monitoreo va a determinar futuros cambios que se deba a hacer al software como el mantenimiento por arreglos de fallas o la evolución para agregarle nuevas funcionalidades que permitan mantener vigente a la aplicación o darle un mejor uso.

La integración de todas las tareas en un ciclo continuo, permite a las empresas que desarrollan sistemas, sacar partido de las oportunidades del mercado de forma más rápida y reducir el tiempo para incluir respuestas de clientes, mejorando la experiencia de los mismos con el software que utilizan.

Veamos los principales problemas que se presentan en el ciclo de vida de una aplicación.

Integración de cambios



Salvo casos excepcionales, el desarrollo de software es una tarea de equipo.

Uno de los principales problemas que hay que resolver es la integración de los cambios que cada desarrollador le hace a la aplicación.

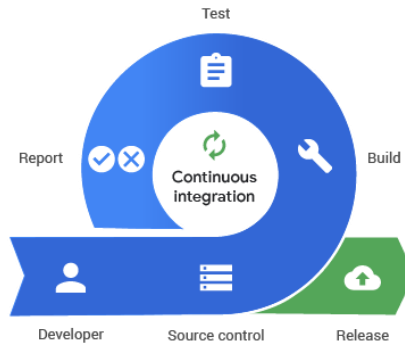
Cada desarrollador trabaja desarrollando una funcionalidad, en un ciclo en el que codifica y testea hasta que cumple su objetivo.

Ahora bien, ¿qué pasa cuando queremos integrar el trabajo de esos desarrolladores, de forma que las funcionalidades en las que trabajaron se incluyan en la siguiente liberación de la aplicación?

Ambos van a intentar subir sus cambios a GeneXus Server pero seguramente van a haber muchos conflictos y más si estos desarrolladores trabajaron de forma independiente por un largo periodo de tiempo. Quizá ambos definieron objetos con el mismo nombre y diferente propósito, o capaz alguno modificó objetos existentes quitando funcionalidades que otro desarrollador cuenta con ellas.

Integración Continua

- Integración frecuente de los cambios, al menos una vez al día
- Verificación de la integración en forma automática



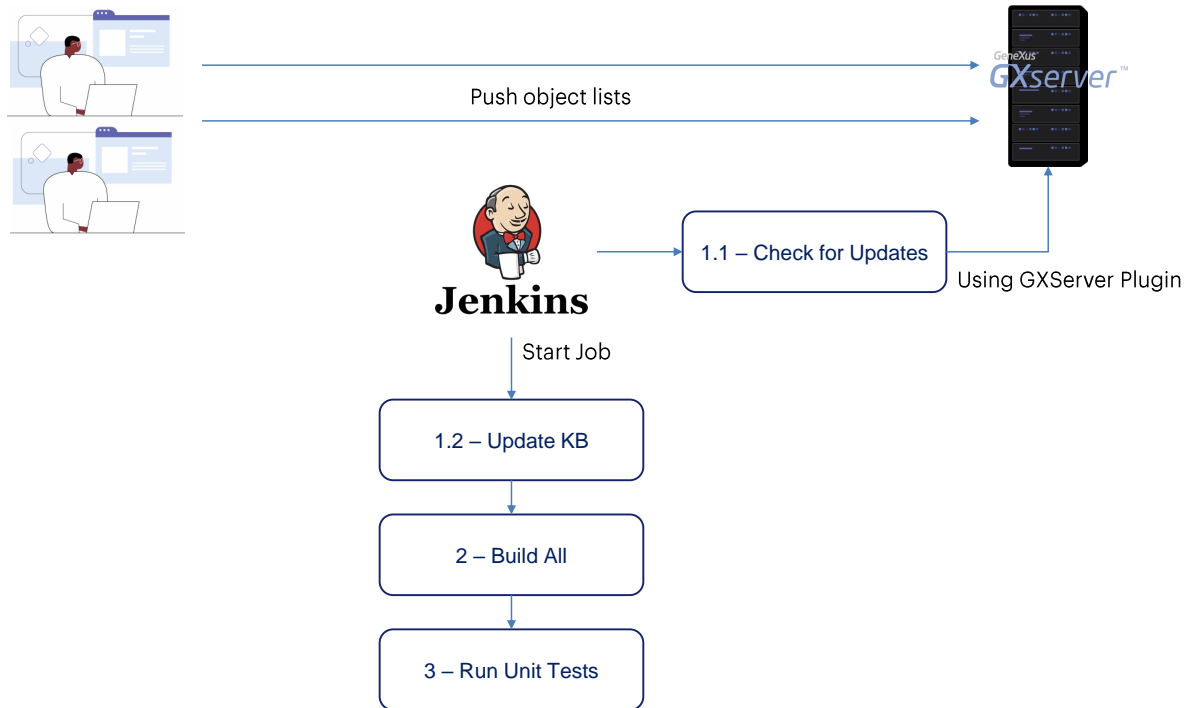
La idea de integración continua es que el trabajo se integre de forma frecuente, si es posible más de una vez al día, de manera de evitar conflictos y en caso de que éstos aparezcan, sean pequeños y fáciles de arreglar.

Una vez que se produce la integración, se debe armar y testear la aplicación en forma automática de forma de detectar los errores de integración lo antes posible. El testeo automático también es útil para validar que no dejó de funcionar algo que ya estaba andando bien, lo que se denomina test de regresión.

Herramientas para la integración continua



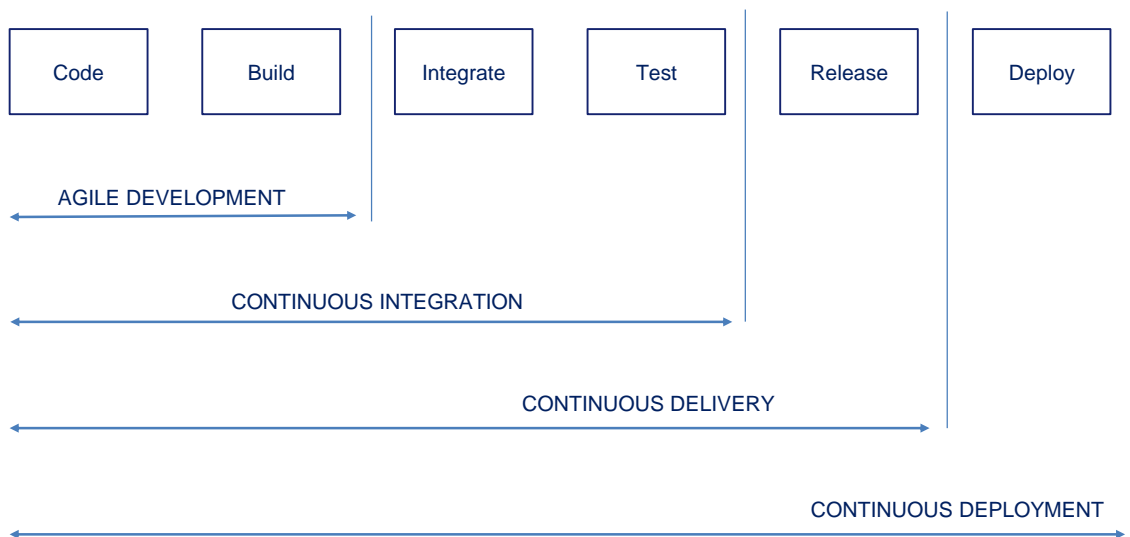
Para esto, vamos a usar GeneXus Server y Jenkins que es un motor de integración continua y es la herramienta que va a orquestar todo el proceso.



En cada commit que haga cada desarrollador, se va a realizar como mínimo el siguiente proceso automático: Actualizar la KB, Build All y ejecutar los tests.

Si todo va bien el build es exitoso y se puede seguir adelante, en caso contrario hay que detener el proceso y resolver el conflicto.

Etapas de ciclo de vida de una aplicación



Estas son las etapas del ciclo de vida de una aplicación.

Si nuestro proceso es capaz de acelerar el proceso de desarrollo y construcción, tenemos una metodología de desarrollo ágil.

Cuando agregamos a nuestro proceso la automatización de la integración y validación es que tenemos Integración Continua

Si a eso le agregamos la automatización del armado del reléase, de forma que luego publicar a producción se pueda hacer con un clic, es que llegamos a Entrega Continua

Cuando nuestro proceso está tan aceitado y automatizado que podemos llegar a poner en producción de forma automática, es que tenemos Puesta en Producción Continua.

Creación y monitoreo de procesos de Integración Continua

JUnitStable - GeneXus 17

File Edit View Layout Build Knowledge Manager Window Tools Test Help

Java Environment Release Android 17.1-SNAPSHOT

Preferences

JUnitStable

- Java Environment
 - Back end
 - Front end
 - Deployment
- Team Development
- GeneXus Cloud
- Patterns
- Workflow

Start Page Team Development

Commit Update History Activity Versions **Continuous Integration**

Integration Pipelines Create

Status	Name	Version	Environment	Run	Last Run	Next Run
Failure	JUnitStable	JUnitStable	Java Environment	69	9/10/2020 17...	12/10/2020 10:31:51
Success	JUnitStableNew	JUnitStable	Java Environment	15	9/10/2020 17...	12/10/2020 17:26:48

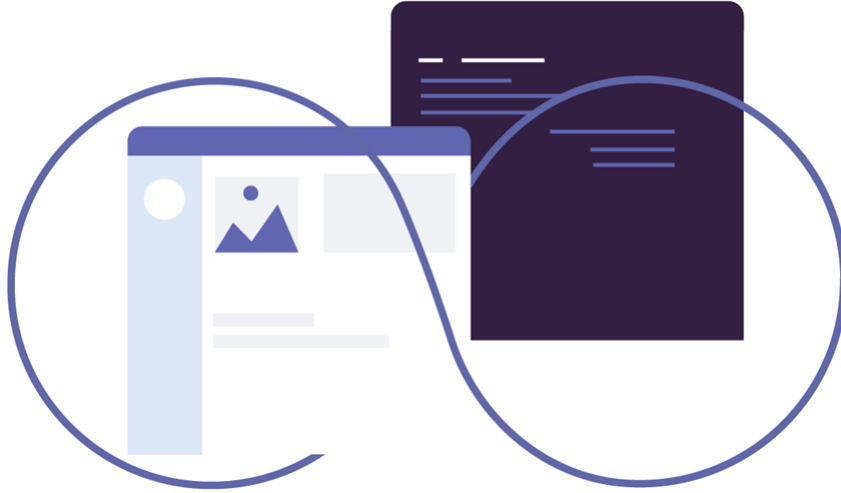
Pipeline Activity

Run	Status	Run Date	Duration
15	Success	9/10/2020 17:26	00:06:05
14	Unstable	6/10/2020 20:19	00:01:01
13	Failure	6/10/2020 20:18	00:00:00
12	Failure	6/10/2020 19:38	00:00:00
11	Success	6/10/2020 18:11	00:00:48
10	Failure	6/10/2020 18:08	00:00:19

Es posible crear y monitorear procesos de integración continua (pipelines) desde el IDE de GeneXus y la consola de GeneXus Server.

Una vez que subimos la KB a GXserver, si vamos a Knowledge Manager / TeamDevelopment disponemos de la solapa "Continuous Integration" en la que poder crear pipelines, ver sus ejecuciones, qué commits entraron en cada una, el resultado y el log completo de cada ejecución.

Proceso DevOps



Y cuando ya incluimos en nuestro proceso la operación y monitoreo de la aplicación en producción, de forma que se genere información que retroalimenta el plan de qué funcionalidades conviene desarrollar después, es que llegamos a tener un proceso DevOps.

Más información:

<https://wiki.genexus.com/commwiki/servlet/wiki?40706>

Por más información sobre el uso de la metodología DevOps en GeneXus,
visite el siguiente link del Wiki:

<https://wiki.genexus.com/commwiki/servlet/wiki?40706>

GeneXus™

training.genexus.com
wiki.genexus.com
training.genexus.com/certifications