



ESTRATEGIAS DE CONTINUOUS DEPLOYMENT

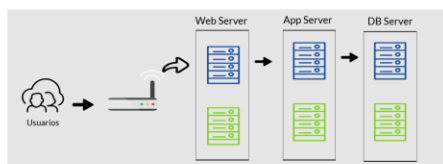
INTRODUCCION

En Continuous Deployment se apunta a minimizar el tiempo que transcurre entre el fin del desarrollo de una nueva versión del sistema y el momento en el cual esta nueva versión comienza a ser usada por los usuarios, buscando automatizar lo más posible la implementación de una nueva versión del sistema ya lanzado.

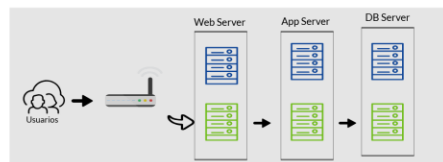
Existen numerosas estrategias, pero hay dos que vamos a destacar: el Blue-Green Deployment y el Canary Deployment. El objetivo del poster es poder analizar como se implementan y poder comprar sus ventajas y desventajas mas significativas.

BLUE - GREEN DEPLOYMENT

Esta estrategia mantiene dos ambientes producción (azul y verde) y sólo uno de ellos se encuentra "en vivo", mientras que el otro se usa para el desarrollo de la próxima versión del sistema.



Ya desarrollada y probada la versión en el ambiente Green, se procede a redirigir el tráfico hacia ese ambiente.



Finalmente, si la versión en el ambiente Green es estable, se procede a apagar el ambiente Blue para utilizarlo como base del desarrollo futuro. La versión Blue se utiliza como backup/rollback en caso de inestabilidad en Green.

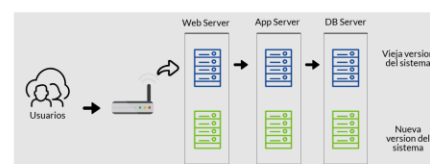
METODO

A través de una tabla comparativa podemos resaltar las ventajas y desventajas de cada estrategia y sus usos, para determinar cual es mejor implementar en software que requiera una implementación continua.

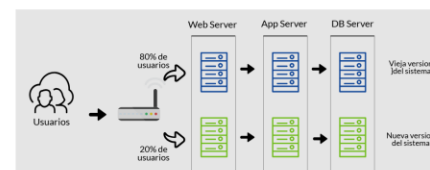


CANARY DEPLOYMENT

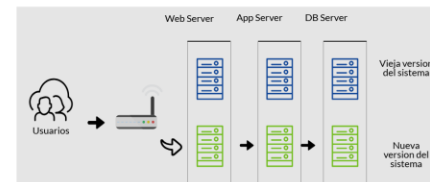
Realiza una migración particionada de usuarios a los distintos ambientes de despliegue.



Una vez desarrollada y probada la nueva versión, se redirige un pequeño conjunto de usuarios selectos (aproximadamente el 15%).



El proceso de migración se completa cuando se garantiza que el ambiente con la última versión del sistema es estable.



COMPARACION

- Migración del sistema completa
- Se utilizan solo dos ambientes
- Es simple de implementar
- Tiene pocos problemas de compatibilidad
- Se enruta al conjunto de usuarios de una vez
- Se pierden todas las transacciones en las transiciones de ambiente



- Migración del sistema por partes
- Se utilizan múltiples ambientes (no tener mas de 3)
- Es complejo de llevar a cabo.
- Da lugar a más problemas de compatibilidad
- Se enruta a diferentes grupos de personas.
- Se pierden una pequeña porción de transacciones en la migración de usuarios



CONCLUSION



No podemos decir que una estrategia sea mejor que la otra, su elección dependerá de las necesidades del sistema.

Si lo que buscamos es simplicidad y no es crítico que el sistema funcione en los tiempos de transición desde la etapa desarrollo y testing al de uso del mismo, es recomendable utilizar Blue-Green Deployment. En su lugar, si necesitamos que el sistema esté disponible siempre para el uso de los usuarios, aunque sea más complejo y costoso de implementar, es conveniente aplicar Canary Deployment.

Por lo general, Blue-Green Deployment es utilizado en empresas pequeñas, mientras que el Canary Deployment para software de uso masivo y de transacciones importantes.

Bibliografía:

- <https://martinfowler.com/bliki/BlueGreenDeployment.html>
- <https://martinfowler.com/bliki/CanaryRelease.html>
- <https://rollout.io/blog/deployment-strategy/>
- <https://opensource.com/article/17/5/colorful-deployments>