Estrategias de despliegue

¿Evaluando el modelo en prod?

- La evaluación del modelo en producción es fundamental para asegurar su correcto funcionamiento.

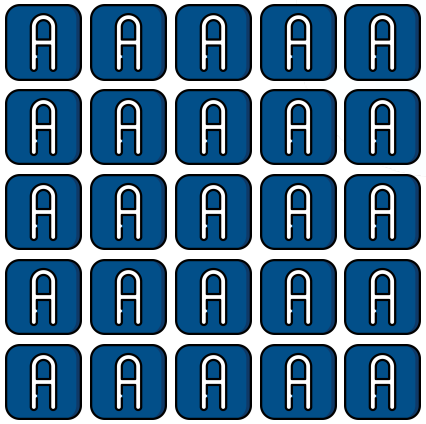
- Existen diferentes estrategias de despliegue que permiten evaluar el desempeño del modelo en un entorno real sin afectar a los usuarios.

¿Evaluando el modelo en prod?

- Para los siguientes escenarios considera que tenemos dos versiones de un modelo: **A** y **B**
  - A es el que está actualmente en producción
  - B es una versión candidata a reemplazar a A.

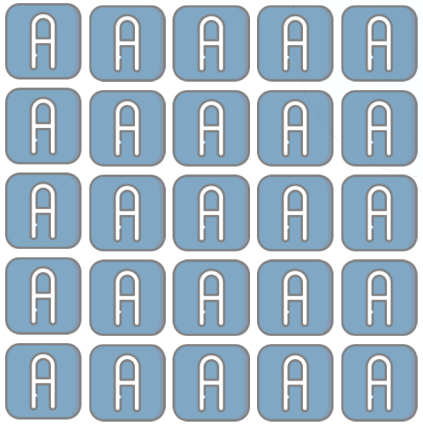
Recreate strategy

- Es fácil y sencilla
- Terminar todas las instancias de la aplicación A y luego iniciar las instancias de la aplicación B



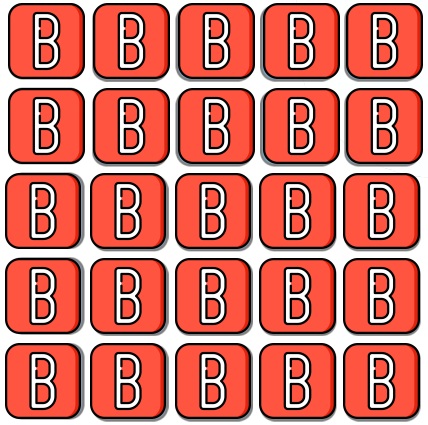
Recreate strategy

- Es fácil y sencilla
- Terminar todas las instancias de la aplicación A y luego iniciar las instancias de la aplicación B



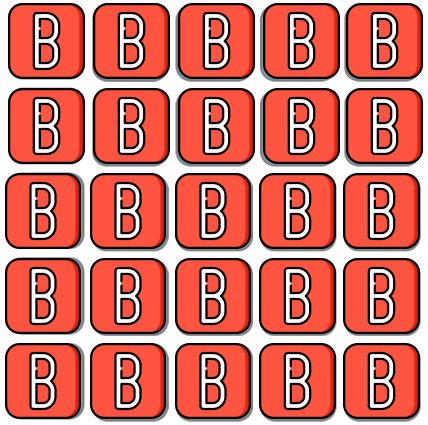
Recreate strategy

- Es fácil y sencilla
- Terminar todas las instancias de la aplicación A y luego iniciar las instancias de la aplicación B



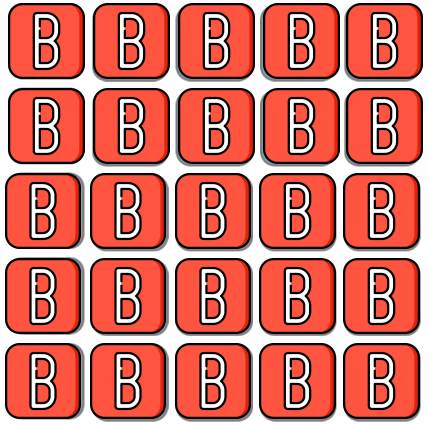
Recreate strategy

- Es sencilla porque no hay complicaciones de redirección de tráfico o análisis en tiempo real
- Sin embargo, es muy disruptiva para los usuarios y cualquier problema afectará a todos los usuarios



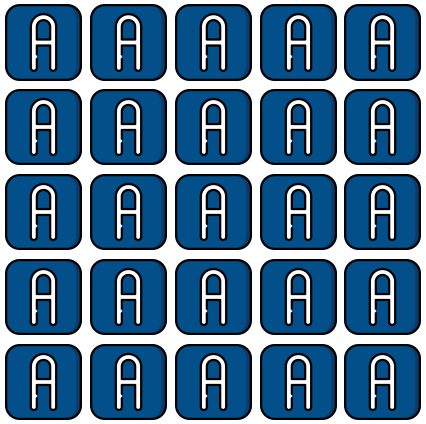
Recreate strategy

- Es sencilla porque no hay complicaciones de redirección de tráfico o análisis en tiempo real
- Sin embargo, es muy disruptiva para los usuarios y cualquier problema afectará a todos los usuarios



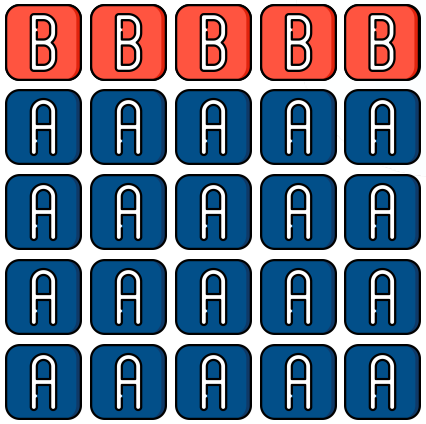
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A



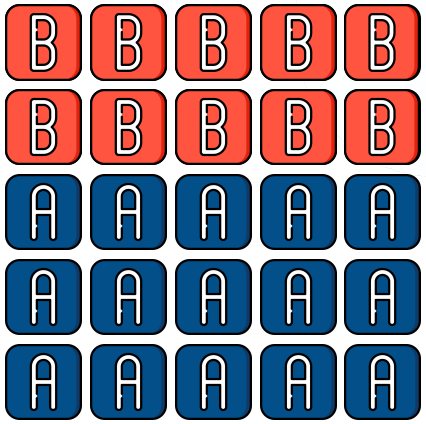
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A



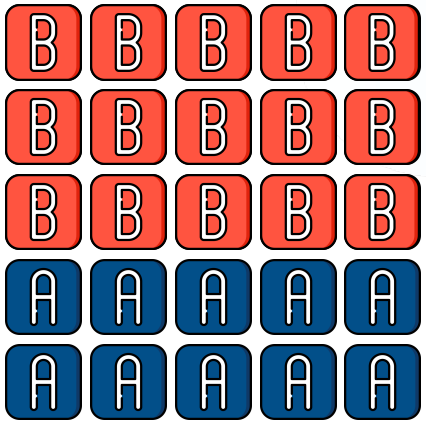
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A



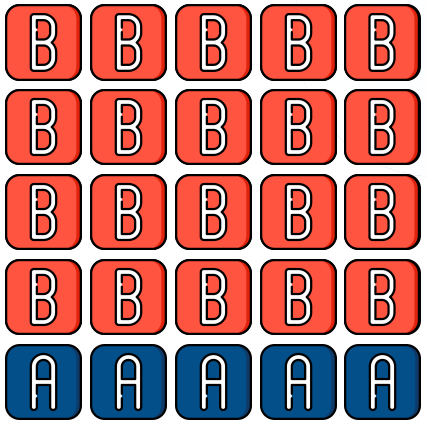
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A



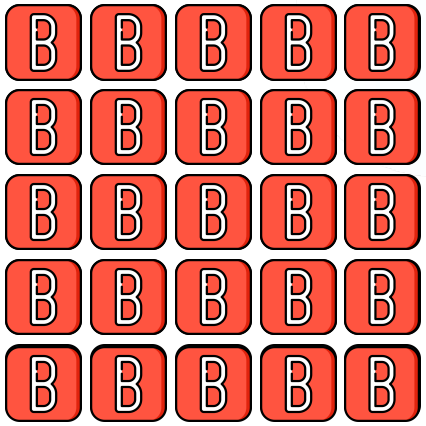
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A



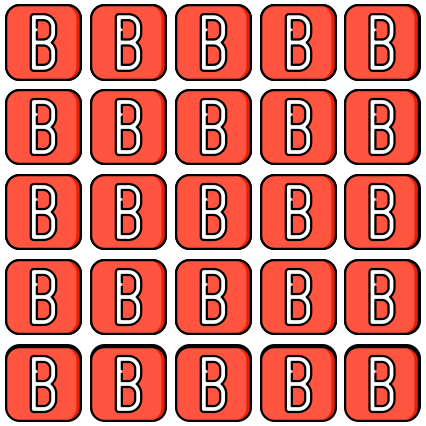
Rolling/ramped update

- Consiste en desplegar lentamente instancias de B y decomisionar instancias de A
- Permite realizar actualizaciones sin periodos de inactividad



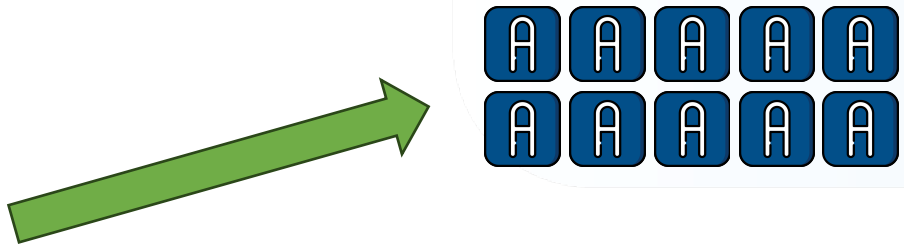
Rolling/ramped update

- Permite encontrar errores tempranamente
- No existe control sobre el tráfico que termina en cada versión del modelo



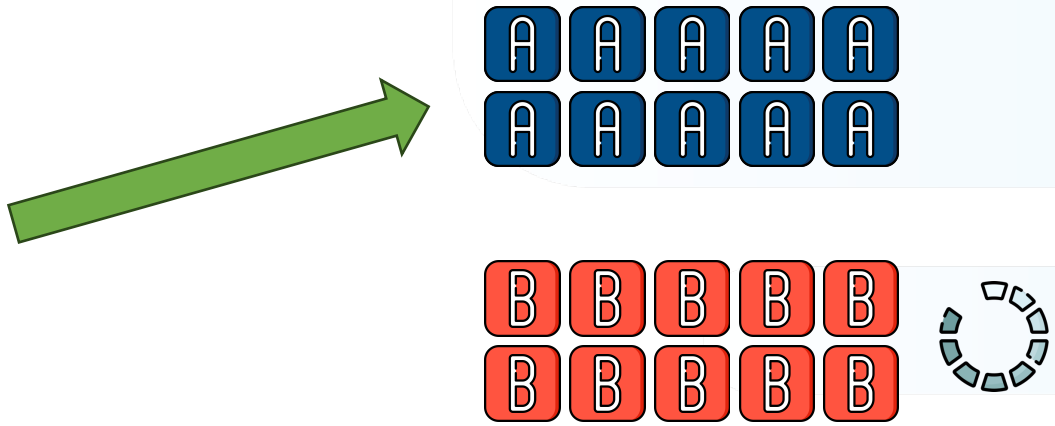
Blue/Green deployment

- Implica poner en paralelo los dos servicios
- El cambio de servicio se hace al 100%



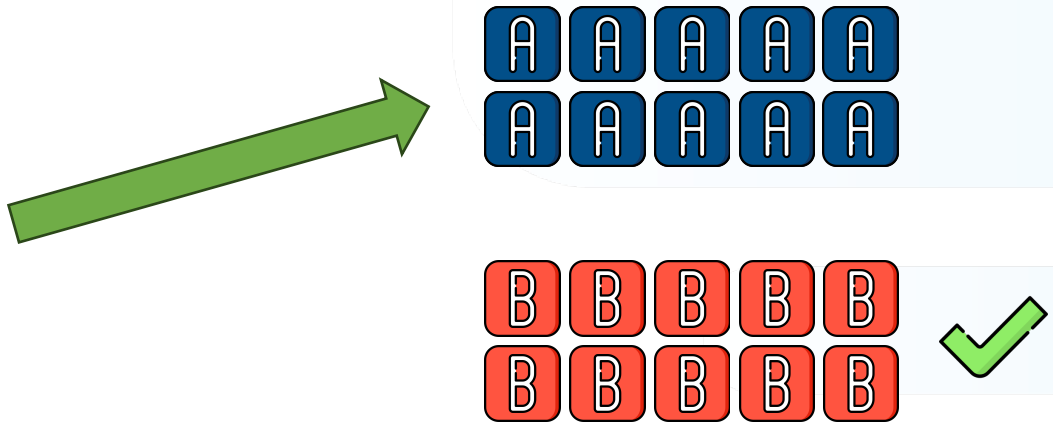
Blue/Green deployment

- Implica poner en paralelo los dos servicios
- El cambio de servicio se hace al 100%



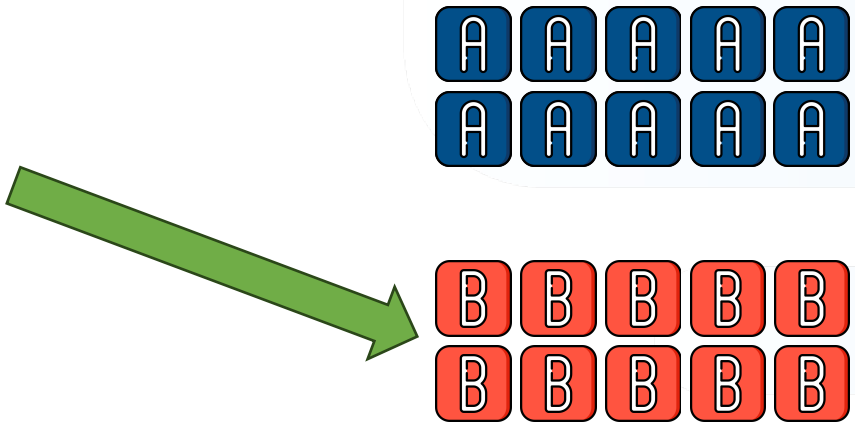
Blue/Green deployment

- Implica poner en paralelo los dos servicios
- El cambio de servicio se hace al 100%



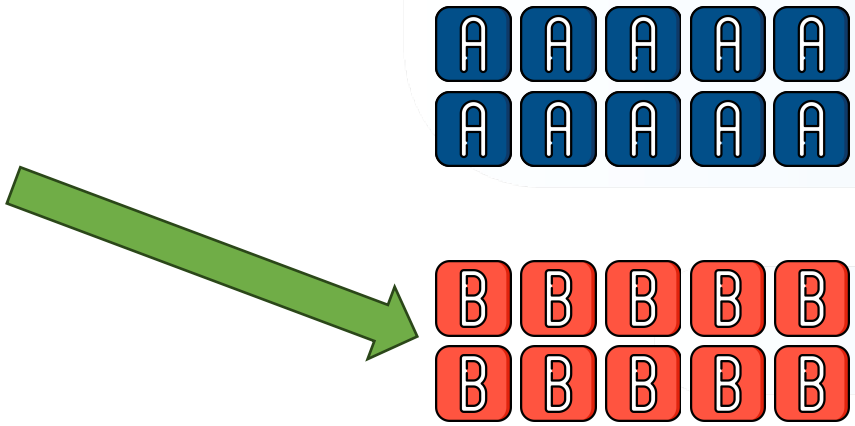
Blue/Green deployment

- Implica poner en paralelo los dos servicios
- El cambio de servicio se hace al 100%



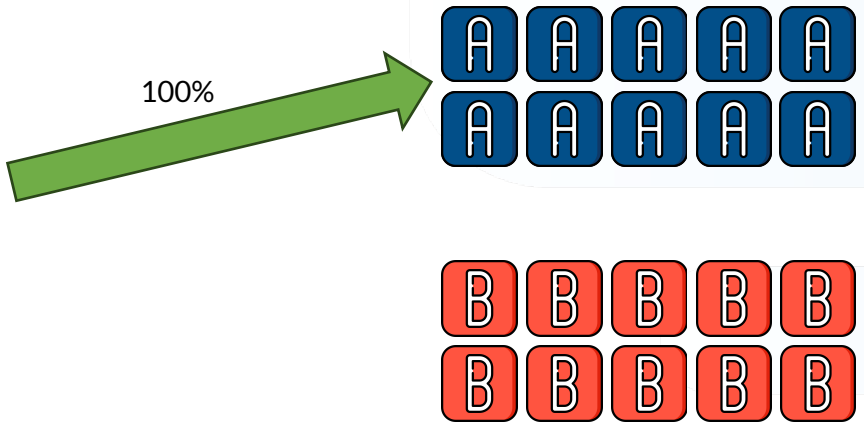
Blue/Green deployment

- Requiere duplicar la infraestructura, pero permite revertir a una versión anterior
- Es costoso, ¡duplicas la infraestructura!



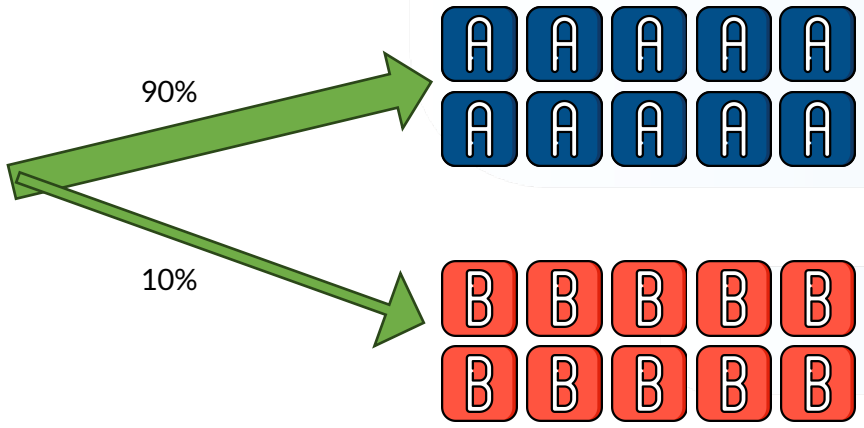
Canary deployment

- Cambiar gradualmente el tráfico de una versión a otra, usualmente por porcentajes.
- Doble infraestructura, probada poco a poco



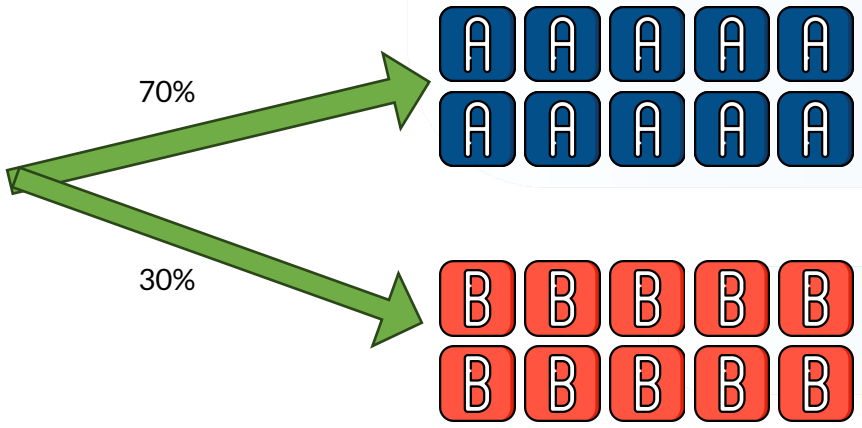
Canary deployment

- Cambiar gradualmente el tráfico de una versión a otra, usualmente por porcentajes.
- Doble infraestructura, probada poco a poco



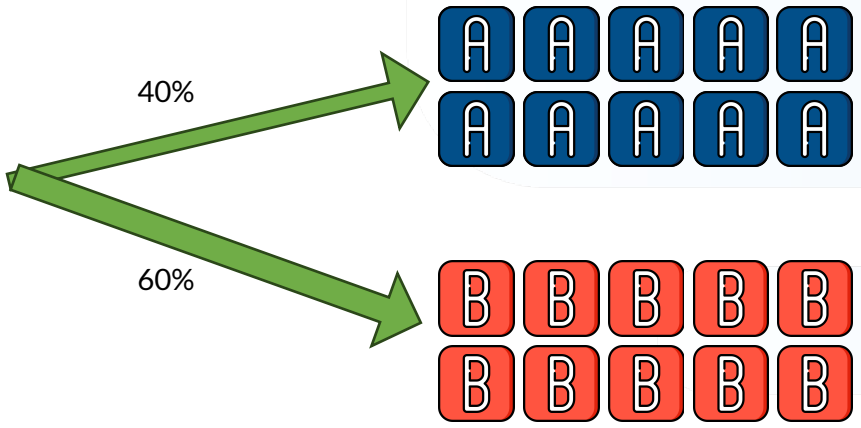
Canary deployment

- Cambiar gradualmente el tráfico de una versión a otra, usualmente por porcentajes.
- Doble infraestructura, probada poco a poco



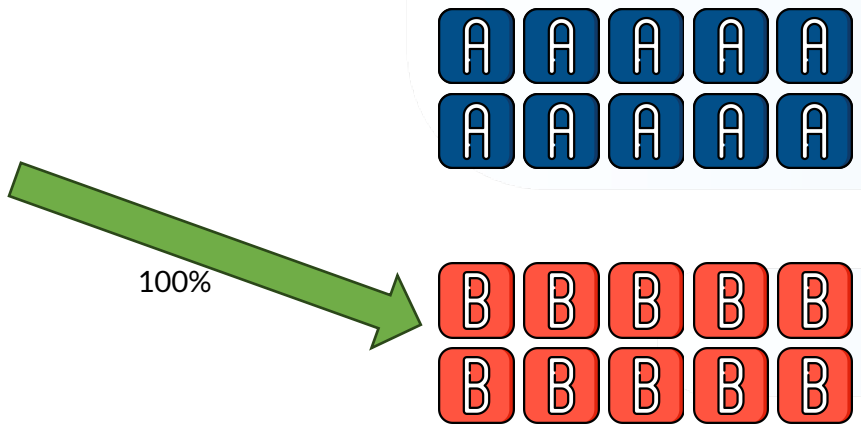
Canary deployment

- Cambiar gradualmente el tráfico de una versión a otra, usualmente por porcentajes.
- Doble infraestructura, probada poco a poco



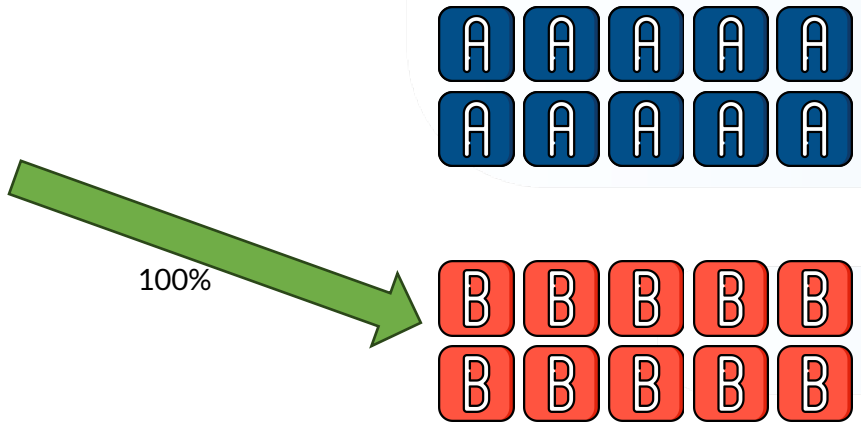
Canary deployment

- Cambiar gradualmente el tráfico de una versión a otra, usualmente por porcentajes.
- Doble infraestructura, probada poco a poco



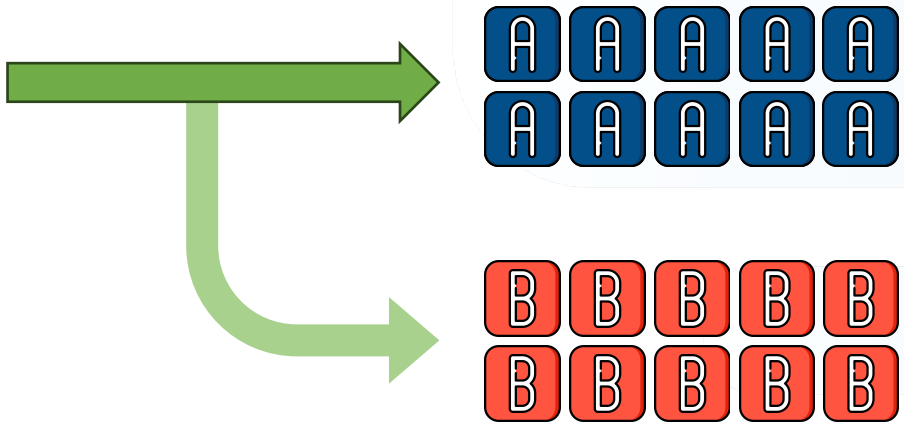
Canary deployment

- Permite que una porción controlada de usuarios pruebe la aplicación y brinde retroalimentación.
- También requiere doble infraestructura



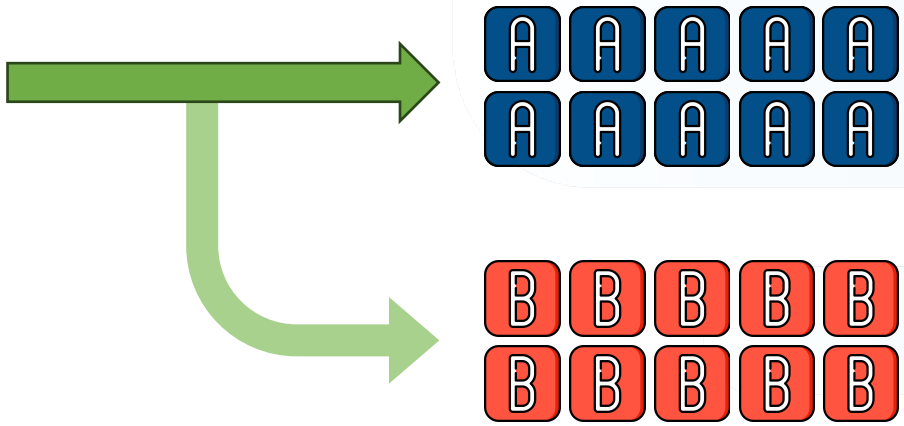
Shadow deployment

- Implica liberar la versión B junto con la versión A, y duplicar las peticiones de A a B para probar cómo reacciona B.
- Permite probar la aplicación sin impactar a los usuarios y garantiza que la versión B funcione correctamente.



Shadow deployment

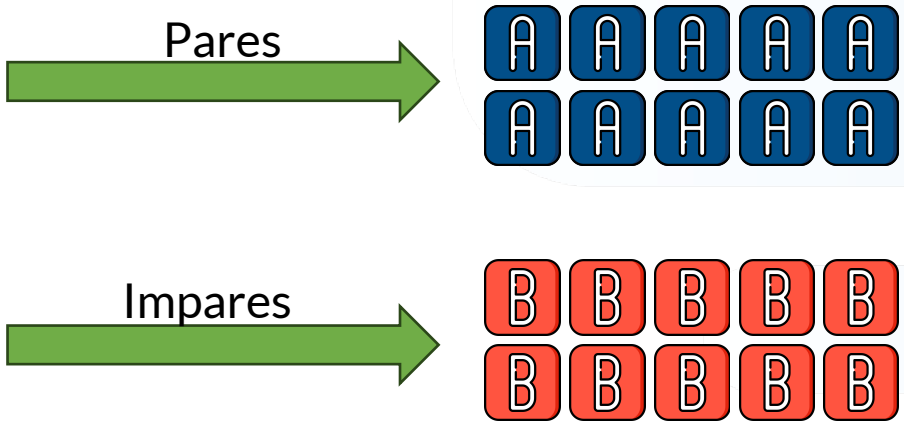
- Requiere duplicar la infraestructura y asegurar condiciones similares para ambos servicios
- Es complejo llevarlo a cabo para evitar duplicados en servicios dependientes



A/B testing

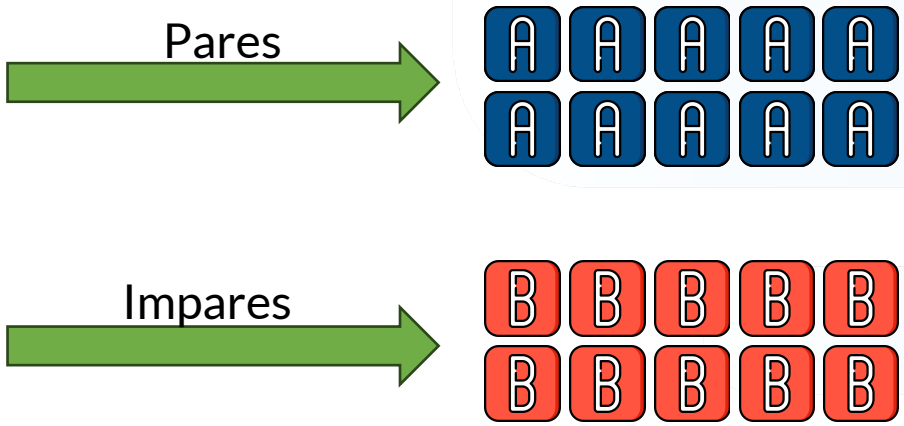
- Los despliegues A/B dirigen subconjuntos de usuarios a diferentes versiones de un modelo para comparar resultados y decidir cuál versión tiene mejor desempeño.

- La clave está en dividir el tráfico de manera determinística para obtener resultados estadísticamente válidos.



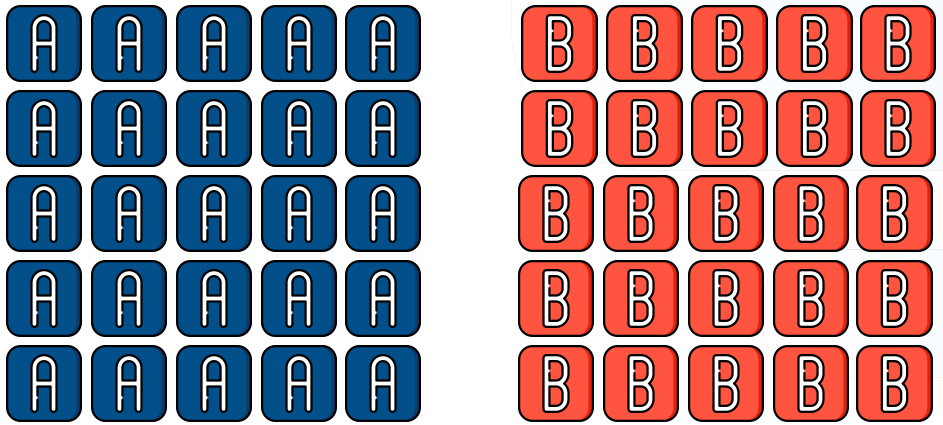
A/B testing

- Permite probar el desempeño funcional de diferentes versiones y tener control sobre la distribución del tráfico.
- Requiere infraestructura adicional, un balanceador de carga inteligente, tiempo y conocimientos especializados, lo que implica costos adicionales.



Recreate strategy

- Es fácil y sencilla
- Terminar todas las instancias de la aplicación A y luego iniciar las instancias de la aplicación B



Estrategias de despliegue

- Cada estrategia tiene ventajas y desventajas.
- La selección de la estrategia está relacionada con el problema a abordar y con los recursos con que cuentas.
- Debe ser una consideración inicial.