



Universidad de Buenos Aires
Facultad de Ciencias Económicas



Estrategias de Despliegue Modernas

AR Tema extractado del libro "**Análisis Funcional de Sistemas y Tecnologías de la Información**" de Aníbal M. Mazza Fraquelli - ISBN 978-987-26981-3-3

Presentación del Tema

Las **estrategias modernas de despliegue** en Tecnologías de la Información (TI) surgen como respuesta a la necesidad de reducir riesgos, aumentar la disponibilidad y acelerar la entrega de valor en entornos digitales altamente dinámicos. A diferencia de los enfoques tradicionales —como el Big Bang o la implantación en paralelo— las estrategias modernas se apoyan en arquitecturas distribuidas, automatización, monitoreo continuo y prácticas DevOps (Development and Operations – Desarrollo y Operaciones).

En organizaciones que operan plataformas digitales, sistemas de comercio electrónico, servicios financieros en línea o aplicaciones móviles, el despliegue ya no es un evento aislado, sino un proceso continuo. En este contexto, la estrategia de implementación se convierte en un componente central de la gobernanza tecnológica y de la gestión del riesgo operativo.

Desde la perspectiva de la administración, estas estrategias impactan directamente en:

- Continuidad del negocio.
- Experiencia del cliente.

- Velocidad de innovación.
- Control presupuestario.
- Mitigación de riesgos reputacionales.
- Optimización del retorno de la inversión tecnológica.

Entre las estrategias modernas más utilizadas se encuentran:

- Blue-Green Deployment.
- Canary Deployment.
- Rolling Deployment.
- Feature Toggles.
- Shadow Deployment.
- A/B Testing.
- Continuous Deployment (Despliegue Continuo).
- Dark Launch.

Cada una responde a contextos organizacionales específicos y presenta ventajas estratégicas particulares.

Desarrollo

1. Blue-Green Deployment

El **Blue-Green Deployment** consiste en mantener dos entornos productivos idénticos:

- Entorno Blue (actual).
- Entorno Green (nuevo).

El sistema nuevo se implementa en el entorno inactivo mientras el sistema actual continúa operando. Una vez validado, el tráfico se redirige completamente al nuevo entorno.

Características:

- Cambio inmediato de tráfico.

- Fácil reversión.
- Minimiza tiempo de inactividad.

Ejemplo:

Una empresa actualiza su plataforma de comercio electrónico. El nuevo sistema se prepara en Green y, tras validación, todo el tráfico se redirige desde Blue a Green.

Ventajas administrativas:

- Reduce riesgo de interrupción.
- Permite rollback rápido.
- Minimiza impacto financiero ante fallas.

Desventaja:

- Duplicación temporal de infraestructura.
 - Mayor costo operativo.
-

2. Canary Deployment

El **Canary Deployment** libera una nueva versión a un pequeño grupo de usuarios antes de extenderla a toda la base.

El nombre proviene del uso histórico de canarios en minas para detectar gases tóxicos antes de afectar a toda la operación.

Características:

- Implementación gradual.
- Monitoreo intensivo.
- Escalamiento progresivo.

Ejemplo:

Un banco lanza una actualización de su aplicación móvil primero al 5% de usuarios para evaluar estabilidad.

Ventajas:

- Reduce impacto global.

- Permite detectar errores tempranos.
- Mejora control de riesgos.

Desventajas:

- Requiere monitoreo avanzado.
- Puede generar versiones simultáneas activas.

Desde la administración, esta estrategia protege la experiencia del cliente y la reputación institucional.

3. Rolling Deployment

El **Rolling Deployment** consiste en actualizar gradualmente instancias del sistema sin detener completamente el servicio.

Características:

- Actualización por segmentos.
- No requiere duplicación completa.
- Mantiene servicio activo.

Ejemplo:

En una infraestructura con múltiples servidores, se actualizan uno por uno mientras los demás continúan operando.

Ventajas:

- Minimiza interrupciones.
- Menor costo que Blue-Green.
- Adecuado para arquitecturas escalables.

Desventajas:

- Posible coexistencia temporal de versiones distintas.
 - Requiere compatibilidad entre versiones.
-

4. Feature Toggles (Interruptores de Funcionalidad)

Los **Feature Toggles** permiten activar o desactivar funcionalidades sin necesidad de desplegar una nueva versión del sistema.

Características:

- Control dinámico.
- Activación selectiva.
- Pruebas controladas.

Ejemplo:

Una nueva funcionalidad de descuentos se activa solo para usuarios premium.

Ventajas:

- Flexibilidad.
- Reducción de riesgos.
- Posibilidad de experimentación.

Desde la administración, los interruptores permiten gestionar lanzamientos estratégicos sin modificar infraestructura.

5. Shadow Deployment

El **Shadow Deployment** consiste en ejecutar una nueva versión del sistema en paralelo, pero sin exponerla a los usuarios finales.

La nueva versión recibe tráfico real, pero sus resultados no impactan en la operación.

Objetivo:

- Comparar comportamientos.
- Detectar inconsistencias.
- Validar rendimiento.

Ejemplo:

Un sistema nuevo procesa transacciones en segundo plano mientras el sistema actual continúa activo.

Ventajas:

- Validación con datos reales.
- Sin riesgo directo para usuarios.

Desventajas:

- Mayor consumo de recursos.
 - Complejidad técnica.
-

6. A/B Testing

El **A/B Testing** consiste en presentar dos versiones diferentes de una funcionalidad a distintos grupos de usuarios para medir resultados comparativos.

Objetivo:

- Evaluar impacto en comportamiento del usuario.
- Optimizar experiencia.
- Tomar decisiones basadas en datos.

Ejemplo:

Un portal digital prueba dos diseños de interfaz para evaluar cuál genera mayor conversión.

Desde la administración, esta estrategia permite decisiones basadas en evidencia y mejora del ROI.

7. Continuous Deployment

El **Continuous Deployment** implica liberar cambios automáticamente a producción tras superar pruebas automatizadas.

Se apoya en:

- CI/CD (Continuous Integration / Continuous Delivery – Integración Continua / Entrega Continua).
- Automatización.
- Monitoreo constante.

Ventajas:

- Alta velocidad de innovación.
- Reducción de ciclos de entrega.
- Mejora competitiva.

Desventajas:

- Requiere alta madurez técnica.
 - Exige cultura organizacional adaptada.
-

8. Dark Launch

El **Dark Launch** libera funcionalidades ocultas al usuario final.

Permite:

- Probar infraestructura.
- Evaluar rendimiento.
- Activar posteriormente para usuarios.

Ejemplo:

Un sistema incorpora un nuevo motor de recomendación que permanece inactivo hasta su activación oficial.

9. Comparación estratégica

Las estrategias modernas comparten características:

- Reducción del riesgo.
- Implementación gradual.
- Automatización.
- Monitoreo continuo.
- Enfoque en experiencia del usuario.

Se diferencian en:

- Nivel de exposición.
- Costo de infraestructura.

- Complejidad técnica.
- Capacidad de reversión.

Desde la perspectiva administrativa, estas estrategias permiten:

- Minimizar pérdidas por fallas.
 - Proteger ingresos.
 - Mantener reputación.
 - Optimizar retorno de inversión.
-

10. Impacto en la gobernanza de TI

Las estrategias modernas requieren:

- Arquitectura escalable.
- Observabilidad (monitorización).
- Gestión avanzada de configuración.
- Cultura DevOps.
- Automatización robusta.

Sin estos elementos, la implementación puede generar mayor complejidad que beneficio.

Desde la administración, su adopción implica inversión inicial mayor, pero reduce riesgos a largo plazo.

Conclusión

Las estrategias modernas de despliegue —como Blue-Green, Canary, Rolling, Feature Toggles, Shadow Deployment, A/B Testing y Continuous Deployment— representan una evolución significativa respecto a los modelos tradicionales. Su principal objetivo es reducir riesgos, garantizar continuidad operativa y acelerar la entrega de valor.

Desde la perspectiva de las Tecnologías de la Información aplicadas a la administración, estas estrategias permiten equilibrar innovación y control, proteger la inversión tecnológica y mejorar la experiencia del usuario.

En entornos digitales competitivos, la capacidad de desplegar cambios de manera controlada y segura constituye una ventaja estratégica fundamental.

Preguntas de autoevaluación

1. ¿Cuál es la principal diferencia entre Blue-Green Deployment y Rolling Deployment?
 2. ¿Por qué Canary Deployment reduce el riesgo reputacional?
 3. ¿Cómo contribuyen los Feature Toggles a la flexibilidad organizacional?
 4. ¿Qué ventajas ofrece el Shadow Deployment frente al despliegue tradicional?
 5. ¿Por qué las estrategias modernas requieren una cultura DevOps madura?
-

Material de Clases

Compilado por **Aníbal M. Mazza Fraquelli** Doctor de la Universidad de Buenos Aires para el uso de sus clases en la Facultad de Ciencias Económicas de la Universidad de Buenos Aires.

Contenidos de esta página

Los contenidos **aquí incluidos integran desarrollos y escritos propios del autor, así como materiales de terceros (documentos, textos, fragmentos, conceptos, imágenes, esquemas, definiciones u otros recursos)**, los cuales son utilizados a título ilustrativo, explicativo o formativo, respetando la normativa vigente en materia de derechos de autor y citando las fuentes cuando corresponde.

La selección, organización, adaptación pedagógica y contextualización de los contenidos constituye un trabajo original del autor, orientado a facilitar los procesos de enseñanza y aprendizaje.

Este material no persigue fines comerciales y su reproducción, total o parcial, queda limitada al ámbito educativo, debiendo preservarse siempre la mención de la autoría y las fuentes originales.

Autorización de uso

Se permite la reproducción, comunicación pública, distribución y utilización total o parcial de los contenidos de su material, en formato físico o digital, con fines exclusivamente educativos, académicos o de divulgación, siempre que se respete la integridad del contenido y se incluya la correspondiente referencia a la fuente y a la autoría.

Las ideas, opiniones e interpretaciones contenidas en este material corresponden exclusivamente al autor.

Queda expresamente excluido cualquier uso con fines comerciales.