

# Siete formas de reducir el desperdicio y acelerar la entrega de software

Cómo el enfoque de la solución DevOps de IBM permite acortar los ciclos de desarrollo de software y aumentar el valor para los clientes

Conozca más



## Contenidos

- 2 Resumen ejecutivo
- 2 Introducción
- 2 Medición del desperdicio y de las cadenas de valor
- 3 Las siete causas clave de desperdicio en el desarrollo de software
- 7 Conclusión

### Resumen ejecutivo

El desarrollo eficiente es una cultura; y la reducción del desperdicio es uno de los resultados. Eliminar el desperdicio mejora la eficiencia operativa, pero lo más importante es que acorta los ciclos de desarrollo y aumenta el valor para los clientes.

Los ciclos más cortos mejoran la innovación, la competitividad y la capacidad de respuesta en el mercado. Además, proporcionan una valiosa oportunidad para el aprendizaje y la mejora continua a los equipos de desarrollo.

Numerosas empresas usan las prestaciones de DevOps de IBM con los principios de desarrollo eficiente para reducir el desperdicio satisfactoriamente. Forrester Consulting analizó a muchos clientes de IBM que usan estas soluciones y determinó que los desarrolladores, los evaluadores y los administradores de proyectos logran ahorros de tiempo de un 30 % a un 50 % en actividades de colaboración y comunicación solamente.

### Introducción

Las técnicas de reducción de desperdicios permiten que los equipos de desarrollo dediquen más tiempo al trabajo productivo, lo que reduce la duración de los ciclos y aumenta la cantidad de código que se puede entregar en un plazo determinado.

El desperdicio en los proyectos de desarrollo de software suele provenir de siete características comunes de dicho desarrollo:

- La espera
- Las entregas y el intercambio de tareas
- El movimiento
- Los procesos adicionales
- Las funciones adicionales
- El trabajo parcialmente terminado
- Los defectos

Este documento analizará cómo el enfoque de DevOps de IBM permite abordar estos siete tipos de desperdicio. DevOps de IBM incluye dos herramientas centrales:

- La solución IBM® Rational® para Collaborative Lifecycle Management (CLM): una plataforma para el proceso de creación de código
- IBM UrbanCode: una plataforma para el proceso de entrega de código

### Medición del desperdicio y de las cadenas de valor

Las cadenas de valor ayudan a comprender cuánto tiempo llevan las tareas y cuánto desperdicio hay en cualquier aspecto del trabajo. A menudo, las siguientes cadenas de valor tienen un volumen significativo de desperdicio:

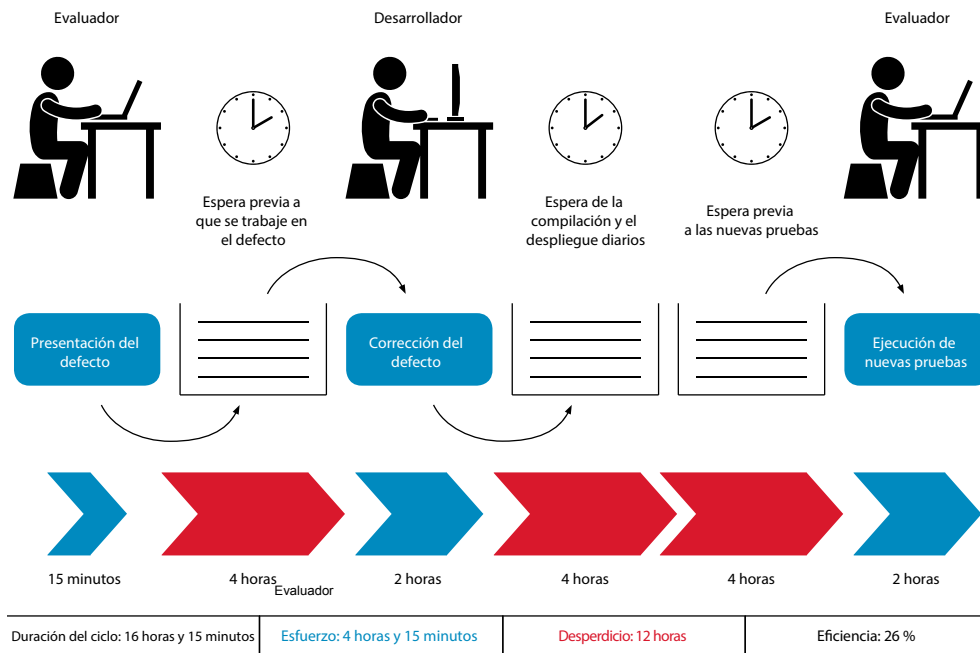


Figura 1. Ejemplo de la cadena de valor de un defecto

- Transformar las nuevas ideas en desarrollo
- Obtener la aprobación de los requisitos
- Buscar la información del proyecto
- Establecer una infraestructura de prueba
- Desplegar una aplicación en el proceso de producción o prueba
- Corregir los defectos

La cadena de valor en la figura 1 muestra cómo una tarea (en este caso, la corrección de un defecto) pasa por una cantidad de personas o equipos hasta que se termina. En cada punto, o bien la tarea está en espera para que se trabaje en ella (el desperdicio se indica con color rojo), o bien se están llevando a cabo acciones para su finalización (el esfuerzo se indica con color azul).

A lo largo de la cadena de valor, tenga en cuenta estas medidas:

- Duración del ciclo = esfuerzo + desperdicio
- Eficiencia = esfuerzo / duración del ciclo

Concentrarse en reducir el esfuerzo y el desperdicio a lo largo de una cadena de valor genera procesos más eficientes y proporciona más valor, más rápido. La reducción del desperdicio debe ser una prioridad más importante que reducir el esfuerzo, porque no solo acorta la duración de los ciclos sino que también aumenta la eficiencia.

Las prestaciones de DevOps de IBM se pueden configurar para calcular el desperdicio y el esfuerzo en cada etapa de cualquier cadena de valor, y estas mediciones servirán para impulsar las mejoras.

## Las siete causas clave de desperdicio en el desarrollo de software

### 1. La espera

#### Esperar a la infraestructura

El tiempo que le lleva a un equipo aprovisionar las máquinas y desplegar el software suele ser la causa de los largos períodos de espera.

Las tecnologías de cloud permiten que los equipos aprovisionen con rapidez una infraestructura de producción, prueba y desarrollo, y que reduzcan potencialmente el tiempo de espera de días a minutos. Las estrategias de cloud también ofrecen acceso a entornos similares a los de producción con anticipación en las etapas de prueba y desarrollo, lo que hace posible eliminar los defectos antes y reducir el riesgo del despliegue.

#### Esperar a que se desplieguen las aplicaciones

El despliegue manual de las aplicaciones puede generar retrasos importantes, en especial, si depende de esperar a que los especialistas del equipo estén disponibles.

Las prestaciones de DevOps de IBM permiten automatizar el despliegue de las aplicaciones. Por ejemplo, una empresa minorista en línea redujo un 90 % el tiempo para cada despliegue usando DevOps de IBM.

#### Esperar a los otros equipos

Los equipos no suelen priorizar el trabajo de manera adecuada. Con frecuencia, no existe una forma sencilla para un equipo de comprender qué aspectos de su trabajo afectan a los demás.

Las prestaciones de DevOps de IBM aumentan la visibilidad de las tareas que están bloqueadas por otras y les permite a los equipos organizarse internamente para reducir el tiempo de espera.

#### Esperar a que finalicen las revisiones

Los autores de los requisitos, los documentos de diseño, los activos de software y el código suelen tener que esperar a que los revisores terminen antes de poder avanzar.

Las prestaciones de DevOps de IBM pueden ayudar mediante la asignación centralizada de estos elementos a los revisores, que posteriormente reciben notificaciones y recordatorios automáticamente. Además, es posible ver el estado de cualquier revisión en cualquier momento.

### 2. Las entregas y el intercambio de tareas

Se produce una entrega cuando un miembro del equipo entrega una parte del trabajo a otro. El intercambio de tareas tiene lugar cuando una persona cambia entre diferentes tareas.

#### Entregas dentro de los equipos de proyectos

Las entregas pueden introducir un volumen significativo de desperdicio por diferentes razones:

- Hacer nuevamente el trabajo a causa de la necesidad de tener que reingresar la información entre herramientas o equipos
- Tiempo de espera que se genera porque los miembros del equipo no saben que una tarea está lista
- Falta de detalles suficientes

- Elementos desactualizados
- Cambios que no se comunicaron eficazmente al equipo

Las prestaciones de DevOps de IBM ayudan a reducir el desperdicio de las entregas entre el desarrollo y las operaciones al permitir que los equipos:

- Automaticen los despliegues.
- Promuevan las aplicaciones entre los entornos.
- Organicen y gestionen los lanzamientos a gran escala.

Con las prestaciones de DevOps de IBM, los miembros de cualquier equipo pueden ver la información y seguir los cambios que se hacen en otros equipos, y así ahorrar tiempo. Esta transparencia reduce el desperdicio en las entregas, porque los miembros de los equipos ven la información que necesitan si tener que molestar a otros para obtener las actualizaciones de estado.

#### **Entregas entre los proyectos y las etapas posteriores**

Al igual que el desperdicio dentro de los equipos de proyectos, también puede existir el desperdicio entre los proyectos, los departamentos y los terceros. Las entregas entre equipos pueden generar muchos desperdicios:

- Una aplicación pasa del equipo de desarrollo a otro equipo para su despliegue.
- Una aplicación pasa a un equipo de soporte con documentación insuficiente, de modo que dicho equipo debe dedicar tiempo a entender cómo funciona.
- Es necesario actualizar una aplicación, pero ya no es posible

encontrar los requisitos originales, el diseño, las pruebas o, incluso, el código.

- Una aplicación debe cumplir con los requisitos que ya se abordaron al desarrollar una aplicación anterior, pero ya no es posible encontrar los diseños, las pruebas ni el código del proyecto anterior.

El enfoque de DevOps que incluye la gestión de activos de software ayuda a resolver estos problemas y reduce el desperdicio mediante el almacenamiento de los grupos útiles de elementos junto con sus metadatos relevantes y su contexto. Con toda la información en un lugar, es fácil encontrar los elementos de desarrollo para los sistemas en mantenimiento. Los equipos pueden identificar rápidamente otros proyectos en los que se resolvieron problemas similares y utilizar de nuevo sus soluciones para ahorrar tiempo.

#### **Intercambio de tareas**

Muchos libros sobre productividad personal hacen hincapié en que hacer muchas tareas a la vez no es la forma más rápida de terminar los trabajos<sup>1</sup>. Los equipos que adoptan la modalidad de desarrollo ágil (Agile) y Kanban siguen este principio para reducir al mínimo la cantidad de tareas en las que se trabaja simultáneamente.

Por lo tanto, en teoría, el mejor enfoque es no sobrecargar a los equipos. De este modo, los equipos no necesitarán hacerse cargo de demasiadas tareas y podrán desarrollar los proyectos en secuencia. No obstante, en la práctica, no siempre es posible.

Si es necesario un intercambio de tareas, las prestaciones de DevOps de IBM pueden ayudar. El desperdicio que se genera en el proceso de informarse acerca de todos los aspectos del proyecto se minimiza porque todos los elementos (procesos comerciales, requisitos, foros de discusión, revisiones, comentarios, pruebas, diseños, código, planes e, incluso, las personas) están vinculados y relacionados. Esta modalidad de trabajo proporciona el contexto necesario y acorta el tiempo requerido para cambiar de una tarea a otra.

Las prestaciones de DevOps de IBM permiten que los desarrolladores tengan muchos cambios de código ejecutándose a la vez, con el mínimo esfuerzo requerido para iniciar o suspender el trabajo en una tarea individual. En lugar de esperar a que un elemento específico esté disponible y listo para su edición, la función de suspensión ofrece una forma de minimizar el desperdicio que surge de la espera y del intercambio de tareas.

### 3. El movimiento

El desperdicio por movimiento surge cuando las personas pasan tiempo físico trasladándose de un lugar a otro; por ejemplo, cuando los miembros del equipo dejan su escritorio para consultar a los expertos en negocios o asistir a reuniones innecesarias.

Las prestaciones de DevOps de IBM reducen el desperdicio por movimiento. Al vincular la documentación y los elementos de los proyectos, y permitir que los miembros del equipo y las partes interesadas los vean fácilmente (por ejemplo, mediante un navegador web), se reduce la necesidad de que las personas se encuentren cara a cara para compartir información. Esta prestación es muy valiosa para los equipos de desarrollo

distribuidos, compuestos por personas que trabajan en diferentes lugares.

En entrevistas con usuarios de DevOps de IBM, Forrester Consulting identificó una reducción del volumen significativo de desperdicio gracias a la capacidad de los equipos de encontrar la información adecuada fácilmente. Un cliente estimó que ahorraron cerca del 70 % del tiempo que antes usaban en las comunicaciones.

### 4. Los procesos adicionales

Los procesos adicionales son aquellos procesos extra que no proporcionan ningún valor al cliente; por ejemplo, crear documentos que nadie lee, recopilar métricas y actualizar planes manualmente, organizar revisiones a las cuales los participantes no responden y otras tareas similares.

Para evitar el desperdicio generado por los procesos innecesarios, evalúe cada paso del proceso en contexto usando el análisis de la cadena de valor o una técnica parecida. Determine si cada paso acelera o demora la entrega de valor para el cliente. Si no agrega valor, considere si es posible eliminar o automatizar ese paso.

Eliminar los procesos adicionales, en especial, a través de la automatización, facilita la implementación de versiones más pequeñas y más frecuentes, la integración continua, las pruebas de integración, la virtualización de pruebas y las pruebas automatizadas. Estas prácticas también permiten reducir el desperdicio ya que acortan la duración de los ciclos y posibilitan un feedback más frecuente.

## 5. Las funciones adicionales

Las funciones adicionales son las funciones innecesarias cuyo diseño exige tiempo y esfuerzo, pero que no ofrecen un valor importante a las partes interesadas del proyecto.

Las prestaciones de IBM DevOps permiten que la empresa colabore en torno a un conjunto de ideas comerciales de alto nivel, les asigne una prioridad, gestione los procesos de aprobación y las destine a la etapa de desarrollo.

Una vez que la idea pasó a desarrollo, la empresa puede observar el progreso en todos los distintos equipos participantes.

Este enfoque ayuda a la empresa a centrarse en las ideas que representan la mejor inversión. Además, reduce al mínimo la cantidad de elementos que se desarrollan, lo que disminuye tanto el esfuerzo como el desperdicio. Mientras tanto, la empresa conserva su agilidad para responder a las cambiantes condiciones.

## 6. El trabajo parcialmente terminado

El desperdicio por el trabajo parcialmente terminado es la cantidad de esfuerzo que se invierte en las funciones que aún no están implementadas.

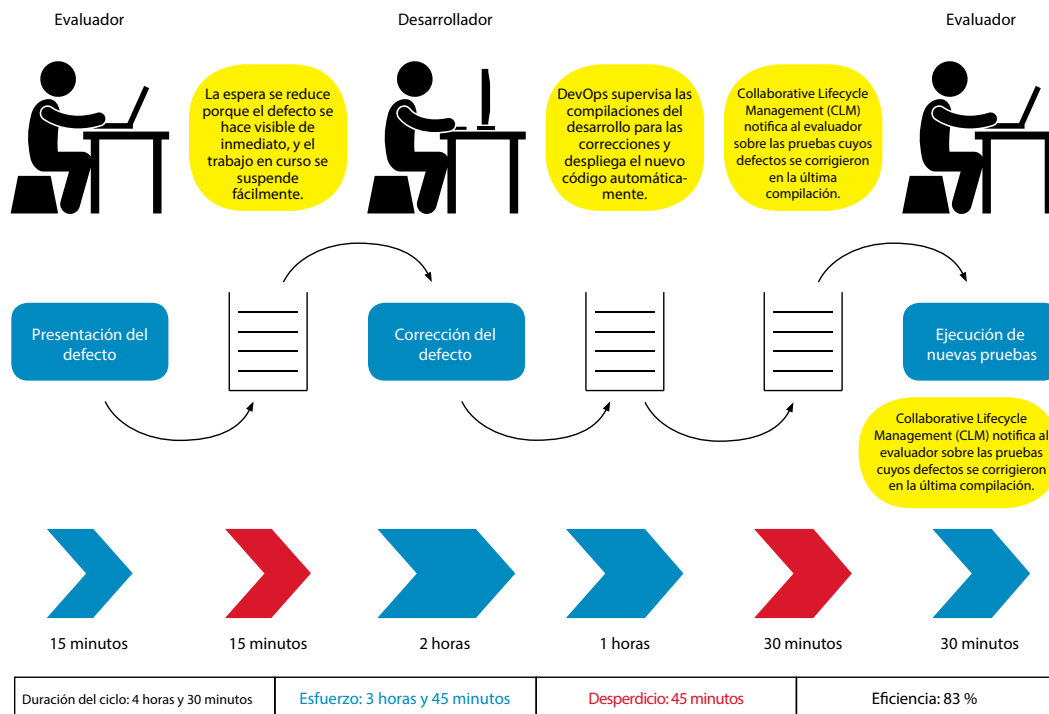


Figura 2. Ejemplo de la cadena de valor de un defecto optimizada por las prestaciones de DevOps de IBM

Usar el enfoque de desarrollo en cascada genera una importante cantidad de trabajo completado parcialmente, porque solo existirá software en funcionamiento en la etapa final de ciclo de desarrollo. No obstante, incluso con un enfoque iterativo, el desperdicio por el trabajo parcialmente terminado puede seguir siendo significativo.

Si el trabajo de un equipo se dispersa en demasiados casos, las pruebas podrían quedar para una etapa muy tardía del ciclo de iteración. Si los defectos se detectan cuando queda poco tiempo para corregirlos, es posible que la iteración termine por no ofrecer nueva funcionalidad.

Por lo tanto, dentro de cada iteración, el enfoque debe orientarse a finalizar un caso tras otro, y el equipo debe trabajar en la menor cantidad posible de casos a la vez. Este enfoque minimiza el trabajo parcialmente terminado y permite que una parte de código terminado se entregue en cada iteración o sprint.

Aquí también las prestaciones de DevOps de IBM pueden ayudar. Los paneles de tareas para planificación del desarrollo ágil se pueden usar para ayudar a los equipos a ver dónde hay trabajo parcialmente terminado y completarlo lo antes posible. Las herramientas de desarrollo conjunto también permiten que un mayor número de desarrolladores trabajen en la misma área de código.

## 7. Los defectos

Los defectos causan dos tipos de desperdicio: el desperdicio por la reparación del defecto y el causado por el software que no funciona correctamente. En enfoque de DevOps de IBM ayuda a evitar estos desperdicios al mejorar la calidad de los requisitos y reducir el tiempo dedicado a buscar y corregir los defectos; o dicho en términos de desarrollo eficiente, al mejorar la eficiencia en la cadena de valor de un defecto.

Cuanto más tarde se detectan los defectos, más caro cuesta corregirlos. Para detectar los defectos y reducir el desperdicio asociado a ellos, es necesario que las pruebas se hagan lo antes posible en el ciclo de vida. Sin embargo, puede ser difícil lograrlo, en especial, en las pruebas de integración, debido a la complejidad que conlleva configurar todas las aplicaciones dependientes.

Con las técnicas de DevOps, como la virtualización de servicios, es posible realizar pruebas de integración con anticipación ya que permiten que las aplicaciones y los componentes se prueben contra versiones virtualizadas (es decir, simuladas) de los servicios

de los cuales dependen, en lugar de los propios servicios reales. Esta práctica es mucho más rápida, menos costosa y, al mismo tiempo, garantiza que la aplicación sometida a prueba se integrará correctamente.

Forrester Consulting informó sobre un banco europeo que aumentó su capacidad de entrega de proyectos un 100 % a lo largo de tres años (con un incremento de 40 a 80 proyectos terminados anualmente) como resultado del uso de las prestaciones de pruebas de integración y virtualización de pruebas de DevOps de IBM.

## Conclusión

Este documento analiza diversas formas en que los enfoques de DevOps de IBM ayudan a los equipos a ser más eficientes e inteligentes. Las prestaciones de DevOps descritas en este documento incluyen las proporcionadas por IBM Rational Asset Manager, IBM Rational DOORS® Next Generation, IBM Rational Focal Point™, IBM Rational Quality Manager, IBM Rational Team Concert™ e IBM UrbanCode.

Utilizar las prestaciones de DevOps de IBM ayuda a eliminar el desperdicio y a automatizar los procesos manuales, lo que permite entregar software de manera más eficiente y con duraciones de ciclos más cortas. Un ejemplo es la figura 2 que ilustra cómo las prestaciones de DevOps optimizan la cadena de valor de un defecto mostrada al principio de este documento.

Además, las prestaciones de DevOps de IBM ayudan a las empresas a conducir sus procesos de desarrollo con mayor eficacia. Al aumentar la visibilidad de todos los aspectos del proyecto de desarrollo para las partes interesadas, DevOps los ayuda a ver el panorama completo, hacer mediciones, supervisar y mejorar continuamente la entrega de software.





---

## Más información

Para obtener más información sobre cómo DevOps de IBM permite crear prácticas de desarrollo de software más eficientes, visite: [ibm.com/devops](http://ibm.com/devops)

© Copyright IBM Corporation 2014

IBM Corporation  
Software Group  
Route 100  
Somers, NY 10589

Documento redactado en los Estados Unidos de América  
Septiembre de 2014

IBM, el logotipo de IBM, [ibm.com](http://ibm.com), DOORS, Focal Point, Rational, Rational Team Concert y `urban{code}` son marcas comerciales de International Business Machines Corp., registradas en muchas jurisdicciones de todo el mundo. Otros nombres de productos y servicios pueden ser marcas comerciales pertenecientes a IBM u otras empresas. Consulte el listado actual de marcas comerciales de IBM en la sección "Copyright y marcas registradas" (en inglés) en [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml)

La vigencia de este documento corresponde a la fecha inicial de publicación. IBM puede introducir cambios en cualquier momento. No todos los productos y servicios ofrecidos están disponibles en todos los países donde IBM opera.

Los datos de rendimiento y los ejemplos de los clientes se incluyen solo con fines ilustrativos. Los resultados de rendimiento reales pueden variar en función de las condiciones de funcionamiento y de las configuraciones específicas. El usuario es responsable de evaluar y verificar el funcionamiento de cualquier otro producto o programa con los productos y programas de IBM.

<sup>1</sup> Por ejemplo: Getting Things Done (Organízate con eficacia), de David Allen, 2002, y Do It Tomorrow (Hazlo mañana), de Mark Forster, 2006



Por favor recicle