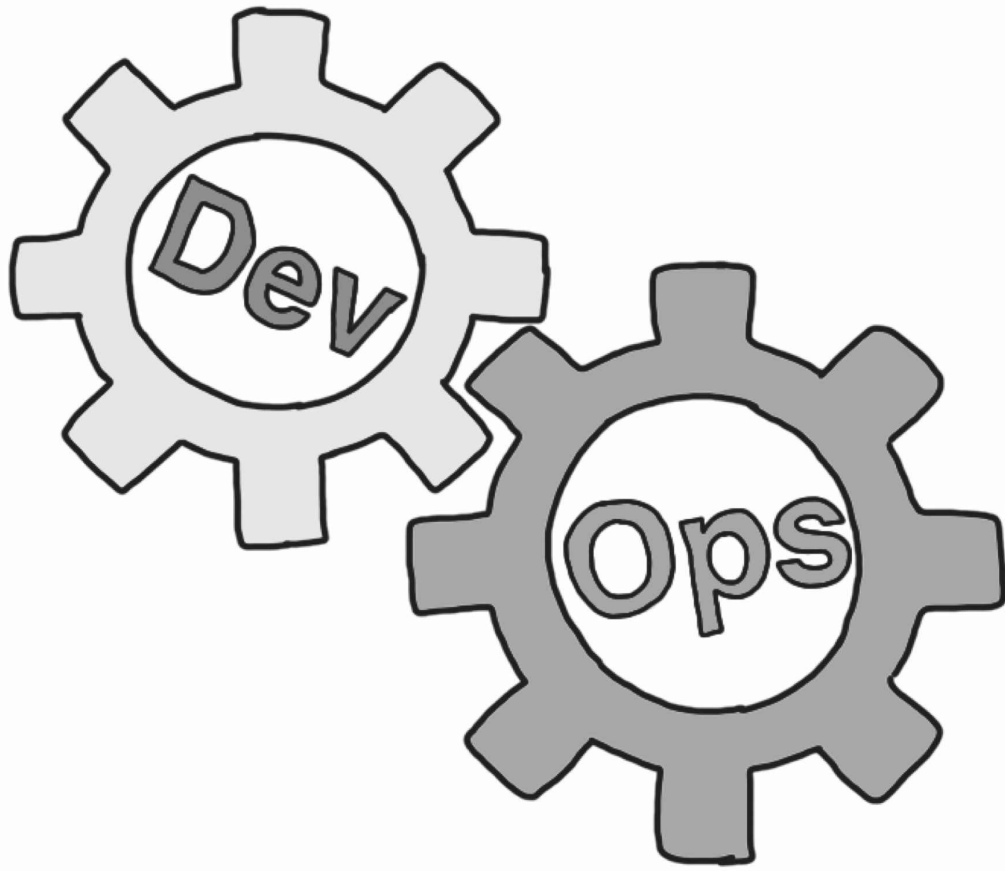


Cultura DevOps

Capítulo 3



¿DevOps?

El término DevOps está de moda, al menos dentro del mundo IT. Solo tenemos que probar a buscar la palabra DevOps en cualquier buscador, para comprobar que aparecen cientos de referencias. El problema para DevOps es que no existe una definición clara que identifique qué es o a qué nos referimos cuando empleamos la palabra DevOps. Empecemos por lo obvio, que además es lo más sencillo, para evitar enredarnos en complicadas definiciones de este esquivo concepto.

DevOps es el juego de palabras que surge al unir las primeras letras de las palabras en inglés:

Develoment Operations

Esta es una primera pista que nos ayudará a entender de qué va DevOps. Se trata de un movimiento que pretende acercar los dos mundos en los que se dividen la mayoría de las áreas de IT, el desarrollo y la operación.

Como término de moda, existe una larga lista de intentos para definir qué es exactamente DevOps. El que no exista una definición establecida, tiene sus pros y sus contras. Entre los contras puedo destacar la confusión que esta falta de definición provoca en las personas que tienen su primer contacto con DevOps. Entre los pros, destacaría que se trata de un concepto abierto que permite que pueda ser adoptado por cualquier organización. Cuando hablo sobre DevOps, suelo utilizar la definición que tiene Wikipedia:

"...se refiere a una metodología de desarrollo de software que se centra en la comunicación, colaboración e integración entre desarrolladores de software y los profesionales de operaciones en las tecnologías de la información (IT). DevOps es una respuesta a la interdependencia del desarrollo de software y las operaciones IT. Su objetivo es ayudar a una organización a producir productos y servicios software rápidamente..."

Wikipedia

Aunque esta definición es bastante completa y explica de manera muy resumida qué podemos entender por DevOps, comente un error al identificar DevOps como una metodología. Este error de Wikipedia es lo primero que debes tener en cuenta cuando te adentras en el mundo DevOps, no es una metodología. Y digo que no es una metodología, porque no existen un conjunto definido de métodos que podamos aplicar para conseguir cumplir con DevOps. Como comentaré más adelante en el libro, DevOps no va de tecnología, ni de metodologías, DevOps va de entender cómo los distintos componentes de un sistema complejo se comunican de una

manera eficiente, para conseguir obtener el mayor rendimiento posible.

Por ahora no sabemos muy bien qué es DevOps, no te preocupes, el propósito del libro no es adoctrinarte sobre qué es DevOps, todo lo contrario, me he propuesto mostrarte cuales son los principios básicos de la cultura DevOps y cómo puede ayudar DevOps a tu organización. Quédate por ahora con esta idea.

DevOps no es una metodología.

DevOps no es una metodología, se trata de un movimiento, que cada uno debe adoptar de la manera que mejor se ajuste a su organización.

**NO ES UNA
METODOLOGÍA**

es una

Cultura

una forma de entender
cómo nos

COMUNICAMOS

DevOps no va de reglas

A los ingenieros nos gustan las normas, necesitamos establecer reglas que unifiquen la forma más óptima en la que se debe trabajar en un problema. Sin las normas estaríamos abocados a repetir una y otra vez errores que otros ya han solucionado y no podríamos saber si estamos aplicando la solución más óptima. Esta es la razón de que se considere a los ingenieros como gente cuadriculada que les gusta seguir los procedimientos.

En IT contamos con una larga lista de normas y metodologías, surgidas de la necesidad de poner algo de orden en un campo de conocimiento que ha tenido un crecimiento enorme en los últimos años. A diferencia de otras áreas de ingeniería, IT han evolucionado de una manera exponencial, lo que ha supuesto un desfase entre la creación de normas y la aparición de nuevas tecnologías.

Este desfase genera problemas a la hora de aplicar al negocio aquellas tecnologías que acaban de aparecer en el mercado. En aquellos casos en los que la metodología no acompaña a la tecnología, se suele optar por un enfoque menos encorsetado y aparecen movimientos con propuestas alternativas. Este es el caso de DevOps, que tiene más de cultura que de metodología, en cuanto que no existe un conjunto de normas y métodos que nos ayuden a transformar un departamento IT en un departamento DevOps.

Para comprender qué es DevOps sería más acertado considerar que está formado por un conjunto de buenas prácticas que *podríamos* utilizar y he puesto en cursiva la palabra *podríamos*, porque realmente este conjunto de buenas prácticas son recomendaciones bastante abstractas y que debemos ajustar a cada departamento, proyecto u organización.

Mi consejo es que intentemos entender DevOps, como un proceso de transformación en la manera en la que los distintos equipos trabajan, colaboran y se comunican, tanto entre sí, como con el resto del sistema. Y esta es la base sobre la que se construyen los principios de la cultura DevOps, no esperes un decálogo de normas y reglas, porque DevOps no va de reglas, va de personas y de incrementar el conocimiento que tenemos sobre el sistema.

#DevOps no va de reglas, va de personas y de incrementar el conocimiento que tenemos sobre el sistema.



Acaba con el muro

Si has trabajado en un departamento de IT, seguro que alguna vez has estado en reuniones en las que gente de desarrollo y de operaciones, se han enzarzado en una discusión por una auténtica niñería. Tal como ocurría cuando éramos pequeños y no le prestábamos un juguete a nuestro hermano, solo porque lo había pedido antes. Esta es la mejor forma de definir la relación entre desarrollo y operación, como la de dos hermanos que están obligados a vivir bajo el mismo techo, pero que si no se comunican están abocados a enfrentarse por cualquier niñería.

Son muchas las causas que originan este enfrentamiento entre el área de desarrollo y el de operaciones, falta de empatía, recelo profesional, inseguridad, afán de notoriedad, rencillas históricas, cultura profesional, pertenencia a un grupo, etc. La relación entre desarrollo y operaciones nunca ha sido fácil, siempre ha existido cierta rivalidad, alimentada muchas veces por los propios equipos. Esta rivalidad ha ido construyendo a los largo de los años un muro que delimita perfectamente las responsabilidades y alcance de cada equipo y que genera una falsa sensación de protección. Nada mejor que un muro para protegerme de los bárbaros, al menos eso pensaron los antiguos emperadores chinos.

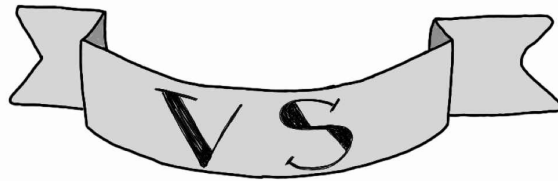
Pero este muro en los departamentos de IT, también ha servido para dificultar aún más la comunicación entre ambos, lo que ha incrementado la altura del muro. De hecho en algunas organizaciones se ha construido un muro tan alto que la relación entre desarrollo y operaciones es casi inexistente.

Es importante que intentemos acabar con este muro para permitir que la comunicación pueda fluir entre ambos equipos, porque a pesar de sus diferencias, la realidad es que sin la colaboración de ambos mundos es difícil para el negocio poder competir en el mercado. Pero derribar el muro no significa unificar, ya que esta es la interpretación que muchos de los detractores del movimiento DevOps enarbolan. No es el objetivo de DevOps eliminar las barreras para fusionar los equipos, sino para que se puedan comunicar, ayudando a la colaboración y cooperación para resolver los problemas de IT, aportando un valor extra y cumpliendo con el principio de que:

$$1 + 1 = 3$$

Las barreras entre Desarrollo y Operaciones terminan impactando negativamente sobre el negocio #DevOps

- PRODUCTIVIDAD -



★ DISPONIBILIDAD ★

¿Qué persigue DevOps?

Es bastante frecuente, encontrar referencias a herramientas DevOps, perfiles DevOps, seguridad DevOps, etc. De hecho mucha gente tiene una percepción errónea sobre el origen del movimiento DevOps, ya que lo relacionan con la utilización de herramientas como el cloud, metodologías ágiles, herramientas de automatización, arquitecturas escalables, etc. Como si DevOps hubiera nacido para resolver un problema de tecnología, todo lo contrario. DevOps aparece en escena para resolver un problema de negocio, en concreto, el problema que tienen la mayoría de las organizaciones para adaptar sus departamentos de IT a las nuevas necesidades de los mercados, que demandan a las compañías un cambio en su modelo productivo. Las compañías deben ser más dinámicas para poder adaptarse a una demanda en continua evolución.

Aunque DevOps intenta resolver un problema de negocio, esto no quiere decir que se trate de una filosofía centrada en el negocio. El objetivo de DevOps es trabajar desde el departamento de IT para poder transformarlo y conseguir incrementar la competitividad de las organizaciones.

Tradicionalmente las áreas de IT se han organizado en dos ramas, cada una de ellas con una responsabilidad concreta. Una rama denominada *Desarrollo*, que es la encargada de crear el software, su función principal es generar el software necesario para dar soporte a los procesos de negocio. La otra rama se denomina *Operación* (también suele denominarse Sistemas, utilizaré Operación por seguir con la nomenclatura DevOps), cuya función es proveer de la infraestructura necesaria para que el software esté disponible y sea accesible a los usuarios.

Podría estar escribiendo decenas de páginas sobre los conflictos que de manera recurrente han azotado la relación entre Desarrollo y Operaciones, pero sobre todos estos conflictos, puedo citar uno, que considero es la madre de todos los conflictos entre Desarrollo y Operaciones, estoy hablando de los objetivos que la organización ha marcado para cada uno de los equipos.

	Función	Objetivo
Dev	Crear	Productividad
Ops	Operar	Disponibilidad

La tabla anterior pone en evidencia el problema real que existe entre las áreas de Desarrollo y Operación, los objetivos de ambas. Aunque estos objetivos deberían ser complementarios, porque así lo necesita la organización, no siempre están alineados. Y de esta desalineación surgen los conflictos entre ambas áreas, conflictos que terminarán impactando en el negocio de la propia organización. Por tanto, si las compañías necesitan ser más competitivas y para ellos deben optimizar

todos sus procesos productivos, la primera acción en un departamento de IT debería consistir en asegurar que los objetivos particulares de cada área, se encuentran totalmente alineados con la organización y lo que es más importante, se encuentran alineados entre las propias áreas.

A la pregunta ¿Qué persigue DevOps? Podemos contestar.

Alinear los objetivos de las áreas de Desarrollo y Operación.

DevOps pretende alinear los objetivos de desarrollo y operación, para que en conjunto coincidan con los de la organización

Las tres vías

DevOps

Primera vía

Entender el sistema como un todo

Segunda vía

Incrementar el feedback

Tercera vía

Experimentación y aprendizaje
continuo

Las tres vías

Aunque DevOps no es una metodología, porque no existe un conjunto de métodos que podamos aplicar, sí existen un conjunto de *buenas prácticas*, las cuales podemos poner en marcha dentro de nuestro departamento IT.

En su libro *The Phoenix Project*, Gene Kim, Kevin Behr y George Spafford, cuentan las peripecias de Bill, un responsable de IT, que se encuentra con el reto de liderar el denominado *Proyecto Phoenix*, el cual es crítico para la compañía en la que trabaja. En el libro se definen tres vías, que han sido adoptadas por parte del movimiento DevOps, como las tres vías básicas sobre la que cualquier organización puede construir su propia cultura DevOps.

- Entender el sistema como un todo.
- Incrementar el feedback.
- Experimentación y aprendizaje continuo.

Aunque parte del movimiento está adoptando estas tres vías, debemos recordar que DevOps es un movimiento, una forma de entender cómo nos relacionamos y puedes adoptar la cultura DevOps sin cumplir con las tres vías anteriores. No tenemos que considerar las tres vías como leyes, sino como consejos que podemos seguir, es nuestro propio *camino de baldosas amarillas* y esta es otra de las claves importantes para entender DevOps. No se trata de conseguir un objetivo al estilo de una certificación ISO que asegura que nuestros procesos cumplen un estándar concreto. Lo importante del movimiento DevOps es crear una cultura corporativa, gracias a recorrer un camino que nos sirva como herramienta de aprendizaje, igual que Dorothy y sus tres compañeros iniciaron un viaje en el que aprendieron cosas sobre ellos mismos y sobre el entorno que los rodea. DevOps es como el camino de baldosas amarillas, si no lo utilizas para alcanzar tus propios objetivos, no sirve absolutamente de nada.

¿Qué metas podemos llegar a conseguir adoptando DevOps como cultura corporativa en nuestro departamento? La lista de beneficios es muy extensa y estarán en función de cada organización y cada departamento de IT, pero a groso modo puedo citar de manera general algunas de las ventajas:

- Colaboración entre los equipos de Dev y los de Ops.
- Incrementar la comunicación entre todas las personas, tanto la comunicación horizontal como la vertical dentro de la jerarquía establecida.
- Generar un ambiente de confianza entre las personas.

- Aumentar la especialización de las tareas.
- Se genera un proceso natural de innovación, al promover el hábito de compartir las ideas con otros, para ayudar a que aparezcan mejoras y nuevos puntos de vista.
- Reducir el Time-to-Market gracias a implantar una mentalidad ágil en la entrega de productos.
- Ahorros relacionados tanto en el proceso de desarrollo como en la explotación.
- Evolución gradual del Sistema, lo que nos ayuda a reducir los tiempos de indisponibilidad.
- Se reduce el tiempo para resolver incidencias.
- Se elimina trabajo repetitivo dentro del sistema.
- Se reduce el tiempo de creación de nuevos entornos.
- Se reduce el tiempo de recuperación ante desastres.

Las tres vías no son leyes, son la base sobre la que debemos construir nuestra propia cultura #DevOps



Mira el sistema como un todo

La primera vía nos propone una transformación de la forma que tenemos de comprender los sistemas de información, para que amplíemos nuestra visión local del sistema, por una visión global que nos permita ver todo el conjunto. Los sistemas de información cada vez son más complejos, ya que al aumento de volumen de información que deben manejar, hay que añadir el incremento en las fuentes de los datos. Y la única manera realmente efectiva para acometer la construcción de un sistema complejo consiste en dividir el sistema en unidades más pequeñas gestionadas por especialistas.

Esta estrategia de reducir los problemas, en problemas más sencillos, que sean más fáciles de solucionar, no es una mala estrategia, el problema aparece cuando la estrategia de dividir en subsistemas más manejables, no va acompañada de una visión global del sistema. Focalizar todo nuestro esfuerzo en un solo punto del sistema, sin tener en cuenta cómo está afectando al resto de subsistemas, es una mala estrategia, ya que nos conduce a trabajar en modo *frontera*, con las consecuencias que este modelo tiene para el rendimiento óptimo del sistema.

Trabajar en modo frontera consiste en que solo me preocupo de qué está ocurriendo en la parte del sistema sobre la que tengo responsabilidad, es decir, sobre mi territorio y no tengo en cuenta qué está ocurriendo fuera de mi frontera. Además si alguien intenta cruzar mi frontera, me ocupo de dejarle claro que está entrado en mi territorio.

El problema de trabajar en modo frontera, es que tarde o temprano, ocurrirá algún problema en alguna de las fronteras y este conflicto impedirá que la información se mueva de manera fluida dentro del sistema, con el consiguiente impacto que tendrá sobre los procesos de negocio. Si por el contrario, adoptamos una actitud más abierta, que nos permita focalizar nuestro esfuerzo en aquellos subsistemas sobre los que tenemos responsabilidades y también entender que somos parte de algo más grande, estaremos incrementando no solo el rendimiento del sistema, sino el conocimiento que podemos adquirir sobre el propio sistema, gracias a interactuar con el resto de personas, cuyas responsabilidades y territorios no coinciden con los nuestros.

Esta es la verdadera razón por lo que es tan interesante para la organización, que todos tengamos una visión del sistema de información *como un todo*. Porque ayuda a incrementar el conocimiento que la organización tiene sobre su propio sistema. Y es desde este incremento del conocimiento propio, desde el que la organización puede construir su estrategia de mejora e innovación, ya que sin un conocimiento profundo, la única forma de mejorar algo es mediante un proceso de prueba y error.

Las compañías necesitan construir sistemas de información cada vez más complejos, que gestionen un mayor volumen de datos y que sean capaces de reducir el tiempo necesario para transformar datos en información. Esto requiere que los sistemas sean lo suficientemente flexibles para amoldarse a las necesidades del negocio. Sin una visión global del sistema, que nos permita tener un conocimiento profundo sobre su alcance y funcionamiento, las áreas de IT no conseguiremos mantener alineados los sistemas de información con las necesidades del negocio.

Trabajar en modo frontera dentro de un sistema terminará perjudicando a la forma en la que información fluye dentro del propio sistema #DevOps

Feed 
 back

Incrementa los flujos de feedback

La segunda vía nos propone que aumentemos nuestros canales de feedback con el sistema, para que podamos obtener información sobre cómo está funcionando y si se mantiene dentro de la zona rendimiento esperado. Pero antes de continuar quiero detenerme en una cuestión ¿qué entendemos por un canal de feedback? El feedback es la información que de una manera u otra obtenemos del sistema y que nos ayuda a conocer cómo está funcionando parte de sus componentes. Podemos tener canales de feedback que no den información sobre los distintos elementos de un subsistema, sobre el uso que realiza un usuario de una aplicación, el grado de satisfacción de los clientes, etc. Podemos tener canales de feedback de cualquier parte del sistema, usuarios, clientes, procesos, software, hardware, etc.

Toda la información recogida por los distintos canales de feedback, debe ser tratada para que tenga una utilidad real para la compañía, ya que no sirve absolutamente de nada desplegar un sistema de monitorización que nos permita medir la ocupación de las CPUs de todas las máquinas, si no somos capaces de relacionar este consumo de CPU con las operaciones que están realizando los usuarios. Por tanto, es necesario procesar toda la información recogida por los distintos canales, para que puedan de alguna manera retroalimentar al sistema.

Existen dos estrategias a la hora de diseñar una arquitectura para nuestros canales de feedback de un sistema:

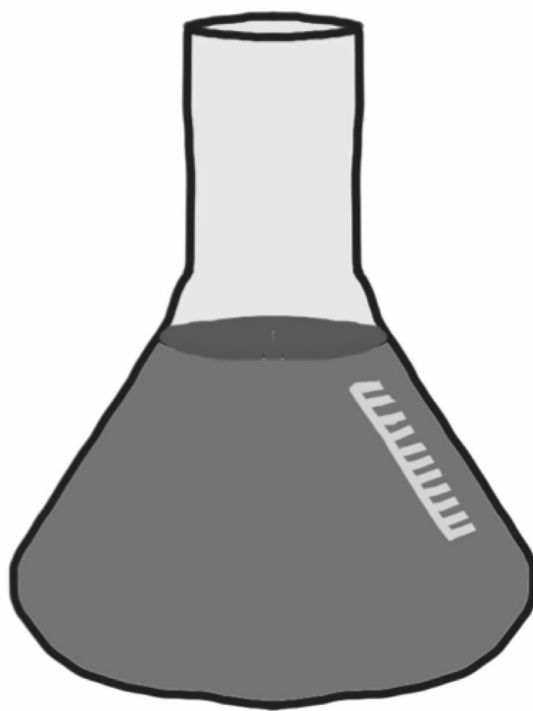
- Analizar de uno en uno, la viabilidad de abrir un nuevo canal, estudiando la calidad de la información que se obtiene del propio canal y el impacto que dicha información tiene sobre el Sistema. Esta visión restrictiva, requiere de más tiempo y su principal inconveniente es que podemos no descubrir la importancia de cierto canal que hemos pasado por alto.
- Abrir todos los canales que creamos útiles, para que sea el propio sistema el que evalúe su utilidad. Esta estrategia es mucho más ambiciosa que la estrategia anterior, requiere de más trabajo, pero nos puede ayudar a explorar aquellos canales que, en un principio pueden quedar ocultos. Presenta el inconveniente de que debemos analizar una mayor cantidad de información y esto puede provocar que entre toda esa información pasemos por alto aquella que es realmente importante.

Independientemente de que elijamos una estrategia u otra, lo importante es que debemos tener una actitud abierta a la hora de diseñar los canales de feedback para nuestro sistema, ya que nunca sabemos dónde puede nacer una nueva fuente de información útil que nos ayude, no solo a reducir riesgos, también a mejorar el rendimiento y/o funcionalidades del sistema.

Al incrementar los canales de feedback, estamos consiguiendo dos objetivos, por un lado, reducir el nivel de incertidumbre sobre el propio ciclo de vida del sistema y por otro lado, estamos incrementado el conocimiento que la organización tiene sobre el propio sistema, su funcionamiento y las posibles mejoras que se pueden aplicar.

Los canales de feedback nos permiten están conectado al sistema para saber cómo se está comportando y cuantificar el desvio sobre el rendimiento esperado.

EXPERIMENTA



Experimenta y aprende continuamente

La tercera vía propuesta rompe con uno de los axiomas que ha imperado en la mayoría de los departamentos de Operaciones IT durante muchos años.

Si funciona, no lo arregles.

Este principio ha servido para evitar muchos problemas y asegurar, en la medida de lo posible, la estabilidad de una plataforma IT, dicha estabilidad se traducía en un aumento de la disponibilidad, que como he comentado en algunos de los puntos anteriores, está relacionada directamente con los objetivos planteados para el departamento de Operaciones.

Aunque no puedo negar los beneficios que ha supuesto para la estabilidad y la disponibilidad, esta forma de entender la operación de los sistemas, también tiene una cara negativa, hablo de los problemas provocados por la obsolescencia de parte de sus componentes. Si entendemos que todos los elementos de un sistema tienen un ciclo de vida y que terminado este ciclo de vida, entran en una fase de obsolescencia, que requiere que los sustituyamos o los actualicemos, es en este momento, en el que pueden aparecer problemas de estabilidad para la plataforma.

Al ser un sistema en el que todos sus componentes, trabajan de manera más o menos coordinada, tener que cambiar uno de estos elementos, puede tener un impacto negativo en otros elementos del sistema con los que interacciona el que estamos actualizando. Este impacto negativo suele desembocar en una falta de rendimiento del sistema o algo peor, la aparición de incidencias fantasmas, lo que terminará afectando a la calidad que perciben los usuarios del sistema.

Por tanto, la tercera vía nos propone que rompamos esta visión tradicional en las áreas de operación, en favor de un nuevo axioma:

Si funciona, mejóralo.

No importa lo bien que funcione un sistema, siempre existe algo que se puede mejorar y en entornos complejos como son los Sistemas de Información, donde existen una cantidad enorme de elementos, siempre podemos aplicar una mejora.

DevOps nos propone un cambio de actitud frente a nuestra visión estática del sistema, para que adoptemos un enfoque más dinámico, que nos permite aprender el propio sistema y que este aprendizaje nos ayude a establecer una estrategia para conseguir evolucionar el sistema en función de las necesidades que demande el negocio.

Plantear una evolución continua del sistema, ayuda a las organizaciones a

disponer de una herramienta flexible que se ajusta en el menor tiempo posible a sus necesidades, con el consiguiente impacto positivo que tiene esta estrategia para el negocio.

Mantener un análisis continuo del Sistema, que nos ayude a identificar aquellas pequeñas mejoras que nos permitan evolucionar del Sistema, pero de manera continuada. De este proceso de mejora continua asociado a la experimentación, nace una consecuencia que es el aprendizaje continuo, lo que nos permite nuevas mejoras inducidas por lo que hemos aprendido y gracias a este proceso cíclico, el sistema evoluciona de manera gradual, evitando llegar al periodo de obsolescencia de sus componentes.

La mejora y el aprendizaje continuo nos ayuda a tener un sistema vivo, que se ajusta mejor a las demandas del negocio #DevOps



DevOps está orientado al negocio

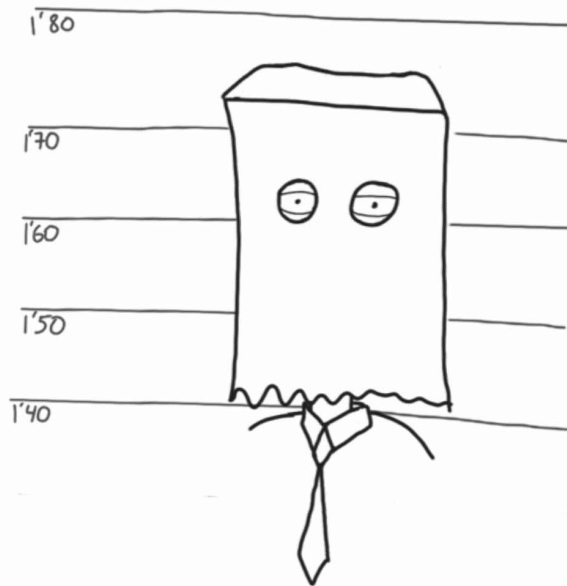
Es bastante frecuente asociar la cultura DevOps con las herramientas disponibles en el mercado. Este error en el planteamiento de la cultura DevOps está consiguiendo inducir la idea que DevOps tiene como objetivo solucionar un problema de tecnología. Por ejemplo, si tenemos un problema en nuestra forma de desplegar aplicaciones, con una herramienta como Jenkins mejoraremos nuestro proceso para desplegar las aplicaciones. Es cierto que herramientas como Jenkins nos ayuda a mejorar el proceso de despliegue, pero realmente el beneficio de utilizar herramientas como Jenkins no lo disfruta directamente el área de tecnología, realmente podemos hacerlo de otra forma, el beneficio es para el negocio, ya que podemos reducir el tiempo de gestión de los despliegues de código y reducir el tiempo total de desarrollo de un producto.

Esta es la razón por lo que se dice que DevOps está orientado al negocio, porque su objetivo no es cambiar la forma en la que trabajamos en tecnología porque tengamos un problema, sino que nos reta a que cambiemos nuestra forma de trabajar para poder impactar de manera más positiva sobre el negocio. Es cierto que el ámbito de actuación de la cultura DevOps es la tecnología, pero los resultados deben tener impacto sobre el negocio. Porque es realmente el negocio el que tiene el problema cuando en IT las cosas no funcionan como se le demanda desde la compañía.

El objetivo es mejorar el sistema de manera que el servicio que ofrece al negocio esté alineado de manera permanente con el negocio. No se trata de aplicar tal herramienta o tal otra, las herramientas son solo herramientas y no tendremos una carpintería por comprarnos un martillo. Esta es la esencia del movimiento DevOps, debemos utilizar las herramientas como lo que son, los vehículos para conseguir cumplir con unos objetivos, que no son el uso en sí de la herramienta, sino la mejora continua del sistema y para cumplir con este objetivo, necesitamos estar orientados permanentemente hacia el negocio.

#DevOps no pretende solucionar un problema de tecnología, sino un problema en el negocio.

SE BUSCA DevOps



Se busca DevOps

Cuando un movimiento o tecnología se pone de moda dentro del mundo IT, como todo, tiene una parte positiva y otra negativa. Lo positivo es que la popularidad del concepto o la idea, tiene un efecto expansivo que le permite llegar a mucha más gente y esto para un movimiento como DevOps es importante, ya que no se trata de una metodología auspiciada por una entidad que se pueda encargar del proceso de difusión. DevOps funciona de una manera más tradicional, de boca en boca. Como no existe una metodología, el proceso de adopción de la cultura DevOps se basa en el trabajo de evangelización tanto interna en las organizaciones, como externa en los distintos eventos que se organizan para discutir sobre la cultura DevOps.

Pero la parte negativa de que sea una palabra de moda, es que suele aparecer en ocasiones en contextos equivocados, lo que genera confusión y suele tener un efecto negativo sobre la impresión que la gente en general pueda tener de la cultura DevOps. Uno de los ejemplos más claros sobre el uso equivocado que se está haciendo del término DevOps, lo encontramos en los cientos o miles de anuncios que aparecen diariamente en cualquier web de empleo. Podemos encontrar anuncios en los que se demandan profesionales con un perfil DevOps. He cogido un anuncio real para el ejemplo:

Se busca perfil DevOps:

- Al menos 4 años de experiencia en puesto similar.
- Conocimientos avanzados de infraestructura.
- Sistemas Operativos: Linux.
- Redes: Configuración de redes virtuales, segmentación, proxies inversos.
- Servidores de aplicaciones y web: Tomcat, JBOSS, Apache, NginX...
- Infraestructura sobre Amazon EC2, S3 y en redes privadas basadas en OpenStack.
- Gestión de la Configuración de Infraestructura con Ansible, Puppet.
- Conocimientos avanzados de scripting bajo Linux: Bash, Perl, Python...
- Herramientas de construcción y despliegue: Bamboo y Jenkins.

Si tenemos en cuenta, que la cultura DevOps pretende promover la

comunicación entre las personas de los equipos IT, que nos invita a que percibamos el sistema como un todo y que deberíamos estar experimentando y aprendiendo continuamente del propio sistema, me surgen varias cuestiones sobre el anuncio anterior:

- ¿DevOps es la persona o la cultura del departamento?
- ¿Soy DevOps si conozco herramientas como Chef o Puppet?
- ¿Por qué no se solicitan capacidades como comprensión, empatía, capacidad negociadora o buena comunicación?

Realmente DevOps no va de Puppet, Bamboo, Jenkins, Python o Linux, por la sencilla razón de que no se trata de un movimiento de herramientas, sino que va de personas y la forma en las que estas personas se comunican y trabajan, para que el sistema funcione. Por tanto, si DevOps va de personas ¿por qué buscamos DevOps en función de las herramientas? La respuesta es que en IT trabajamos con herramientas y por tanto se valora la aptitud frente a la actitud, es cómo hemos funcionado hasta ahora. Creo que sería más correcto reescribir la oferta de la siguiente forma:

Se busca persona con perfil técnico que comparta los principios del movimiento DevOps:

- Comunicación activa, que no le importe hablar con la gente.
- Capacidad para innovar.
- Visión crítica de los procesos y procedimientos.
- Quiera aprender el negocio de nuestra organización.
- Conocimientos técnicos en Puppet, Chef, Jenkins, Python, Linux, Solaris, AWS, etc.
- Tenga curiosidad e inquietud por aprender.
- Pueda trabajar en equipos heterogéneos.
- Tenga una actitud colaboradora en el día a día.
- Sea una persona con empatía.

Contemplando ambos anuncios, ¿cuál crees que se ajusta mejor al propósito de

reclutar gente para un equipo DevOps? El primero se centra en conocimientos técnicos de herramientas, pone el foco en las aptitudes de los candidatos, en los conocimientos que puedan tener sobre una u otra herramienta, pero esto no es DevOps, es ingeniería de sistemas. El segundo anuncio se centra en buscar personas con cierta actitud, aunque también hace referencia a herramientas (no debemos olvidar que vamos a trabajar en un ambiente técnico, en el que hay herramientas, máquinas, lenguajes de programación, etc), el peso de la oferta la tiene la actitud, no la aptitud, del candidato, por una razón, es más fácil enseñar tecnología a una personas que cambiar su forma de ser.

Por tanto, si estás pensando que DevOps puede ayudarte a incrementar el rendimiento de del negocio de tú organización, no empieces poniendo un anuncio como el primer ejemplo, porque lo único que conseguirás, será contratar gente con unos conocimientos extraordinarios en herramientas que quizás su organización no necesita. Empieza por analizar qué perfil de persona encaja mejor en los equipos que quieres formar y cuáles son las actitudes/aptitudes que esta persona deben cumplir, para que los principios del movimiento DevOps puedan arraigar en tu organización.

#DevOps no va de herramientas ni metdologías, va de personas y la relación de éstas con el sistema.

deja
de mirar
tu
ombbligo

IT

Deja de mirar tu ombligo IT

En los últimos treinta años, los departamentos de TI han recorrido el tortuoso camino desde la periferia de las compañías, hasta conseguir convertirse en el núcleo de muchas organizaciones, por no decir todas. Hoy en día, la dependencia que cualquier organización tiene de los sistemas de información es tan grande, que los departamentos de IT se han convertido en áreas estratégicas para el desarrollo del negocio. Como ejemplo, reflexione durante unos minutos, qué harías si mañana al llegar a tu puesto de trabajo, no hubiera PCs, redes WIFI, aplicaciones, servidores, acceso a internet, etc. Seguramente viviría una experiencia al más puro estilo *Walking Dead*, El caos más absoluto se apoderaría de la compañía y sería totalmente inviable poder continuar con la actividad normal.

Como resultado de esta dependencia, las compañías se han visto obligadas a colocar a IT en el núcleo de la organización. Esto ha supuesto un cambio importante en la forma en la que IT se relaciona con el resto de la compañía. Hemos pasado, en estos treinta años, de estar en los sótanos, junto a las cajas de suministro de oficina, atendiendo las necesidades de los usuarios y con un presupuesto ínfimo, que en muchos casos estaba por debajo de las necesidades reales. A escalar niveles dentro de la jerarquía de la organización.

Pero no estoy hablando solo de un cambio en la jerarquía, también ha supuesto un cambio en la propia cultura corporativa de las organizaciones, ya que en IT, hemos pasado de adaptar la tecnología al negocio de la organización, a transformar el propio negocio para que se pueda aprovechar los beneficios de la tecnología. Ha sido este nuevo enfoque el que ha propiciado, tanto el crecimiento de IT dentro de la organización, como la transformación de las compañías para adaptarse a la demanda del mercado.

Aunque es indudable el impacto positivo que los departamentos de IT han tenido para el desarrollo de los nuevos negocios, este movimiento desde la periferia hasta el núcleo ha supuesto también un cambio en la percepción que los departamentos de IT tienen de sí mismos. En muchos casos, esta percepción se ha convertido en una imagen egocéntrica sobre lo importante que somos para el negocio y la necesidad que tiene el negocio de adaptarse a la tecnología y no al contrario, y como resultado de esta visión egocéntrica hemos llegado a pensar que es la compañía la que se debe adaptar a IT y no al contrario. Este enfoque equivocado sobre qué es IT y cuál es nuestra función dentro de las organizaciones, ha provocado en muchos casos, una desalineación entre los objetivos del negocio y los objetivos de los departamentos de IT. Convirtiéndose en muchas ocasiones, en un verdadero lastre para la evolución de la compañía y la competitividad de su negocio.

Uno de los objetivos que debemos establecer a la hora de plantear DevOps

como cultura dentro de nuestro departamento, es poder transformar la visión recíproca que tienen IT y la propia organizaciones. Este proceso de transformación no será un viaje sencillo, pero como todos los viajes comienzan con un paso y este paso debe consistir en generar un cambio en la forma en la que IT se relaciona con el resto de la compañía. El objetivo es que en IT dejemos de mirar nuestro grande, bonito y caro ombligo IT, para que atendamos a las necesidades del negocio. Trabajando en solucionar los problemas reales de la organización, reduciendo los costes e incrementando el rendimiento de los recursos IT.

Si desde IT, conseguimos dejar de mirarnos nuestro ombligo y volvemos a comprender que las compañías no existen como cobertura para nosotros, sino que somos nosotros los que debemos aportar valor al negocio de la organización, explorando nuevas oportunidades para el negocio, aprovechando al máximo los recursos disponibles, ofreciendo la solución que mejor se ajuste a la necesidad del negocio, cambiando nuestra propia percepción sobre éste e intentando alinear nuestra tecnología al negocio de la compañía.

El objetivo es que en IT dejemos de mirar nuestro grande, bonito y caro ombligo IT, para que atendamos a las necesidades del negocio. #DevOps

DEVOPS NO
SIGNIFICA
JEDI



No te vas a convertir en un Jedi

Existe mucha confusión sobre las ideas que están detrás del término DevOps, por la sencilla razón de que se trata de un término de moda y se está convirtiendo en una etiqueta cool que añadir a cualquier producto o metodología. Puedes leer muchas cosas sobre DevOps, desde que es el nuevo paradigma del mundo IT que ha llegado para solucionar todos los problemas de las áreas de tecnología, hasta que se trata de un perfil de super ingeniero, preparado para apagar cualquier fuego que pueda aparecer en la compañía.

DevOps tiene muchas cosas buenas, pero te aseguro que entre todas estas cosas buenas, no se encuentra la de convertirte en Jedi. Si tu compañía necesita un perfil Jedi, es que tiene un problema mucho más grande de lo que la propia compañía piensa y DevOps no va a poder ayudar en nada.

Ya he comentado, que DevOps es un movimiento que debe nacer desde el interior de la propia compañía, con el objetivo de generar una cultura IT propia, que tenga como base las tres vías. Por tanto, no se trata de convertir un perfil en otro tipo de perfil o de contratar personal externo para poner en marcha DevOps. Todo lo contrario, se trata de concienciar a la gente que debe colaborar, comunicarse, trabajar en equipo, eliminar las barreras que existan entre los departamentos, compartir el conocimiento, trabajar por y para el sistema, promover el aprendizaje continuo y todas aquellas acciones que ayuden a mejorar la forma en la que fabricamos el producto de nuestra organización.

Lo realmente importante es hacer que el equipo funcione, al practicar una cultura de la comunicación y la colaboración. Es un error pensar que gracias a DevOps podemos convertirnos en Jedi IT, capaces de resolver cualquier problema que aparezca, ya sea programar código en las aplicaciones, desplegar máquinas virtuales en la nube o automatizar procesos de despliegue de código. Cuidado con esto, porque mucha gente se acerca a DevOps con la intención de conseguir un bonito sable-laser y la esperanza de incrementar tu nivel de midiclorianos.

La cultura DevOps no te va a convertir en un maestro Jedi de IT



Paz, amor y DevOps

Una de las ideas más extendidas sobre la cultura DevOps, es que se trata de un movimiento que promueve la comunicación entre las personas de los distintos equipos, con el propósito de reducir las fricciones que puedan existir, bien a causa de unas deterioradas relaciones personales, bien por una falta de entendimiento entre la forma de repartir tareas y/o responsabilidades en un proyecto. Ya sea por una razón u otra, la realidad es que la calidad de la forma en la que las personas se comunican dentro de una organización, es un factor clave para el buen funcionamiento de ésta.

Pero no nos confundamos, DevOps defiende la necesidad de incrementar la calidad de los canales de comunicación, tanto entre personas, como con el propio sistema y promueve que este proceso, sea un proceso que nazca dentro de los departamentos y alimentado por las propias personas que forman los equipos de IT, para que pueda germinar como el origen de una nueva forma de hacer las cosas dentro de la compañía. Es decir, no debemos entender el movimiento DevOps como un árbitro que se debe encargar de mediar en todos los conflictos que puedan existir y/o aparecer dentro de las áreas de IT.

Es un error caer en la idea de convertir la filosofía DevOps en una especie de “*casco azul*” que se interpondrá entre las personas, para evitar todo tipo de conflictos. Aún más grave es asumir que el objetivo de DevOps sea crear un ambiente idílico de fraternidad entre los distintos equipos, en el que el buen rollo sea la característica predominante. Y una vez que DevOps se ha impuesto como cultura, un arcoíris multicolor se instalará sobre el edificio de nuestra compañía de manera permanente, ya que todo será paz, amor y tranquilidad.

Tan perjudicial es para el funcionamiento de un equipo, el que existan fricciones que impidan una comunicación fluida, como el que se instale en la organización, un ambiente de felicidad y buen rollo ficticio, que impida tener una actitud crítica con la forma en la que se están haciendo las cosas.

Debemos intentar conseguir un término medio, entre la desconfianza y los roces que provocan una mala comunicación entre los equipos y el *colegueo* excesivo que impide a la gente proponer ideas críticas sobre cómo se podría mejorar la forma en la que se está trabajando. Por tanto, es necesario que la relación entre las personas se base en la confianza y la colaboración, sin intentar imponer ningún tipo de cultura del *buen rollo* que obliga a las personas a relacionarse de una manera que no es natural para el ambiente real.

No esperes que aparezca un arcoíris de felicidad sobre el edificio de tu compañía, #DevOps va de mejorar la relación entre las personas.

