

---

# DevOps svelato

Cos'è e perché è in grado  
di rivoluzionare il tuo business

---



**Introduzione**

**Cos'è il DevOps**

**Da dove viene il DevOps**

**Perché non puoi fare a meno del DevOps**

**Cosa significa lavorare secondo DevOps**

**I rischi di un approccio non corretto**

**Il DevOps per il cloud e poi?**

# Introduzione: cos'è il DevOps

Il Paper che hai appena scaricato ha l'obiettivo di introdurre la metodologia DevOps attraverso una definizione chiara ed esaustiva. Vogliamo, poi, illustrarne il modello in una modalità pragmatica che ti permetta di avere un chiaro colpo d'occhio sulla struttura, gli strumenti e il ruolo dei diversi attori coinvolti.

Tra questi ci siete anche tu, i membri del tuo team It e altre risorse della tua azienda, e giocate tutti un ruolo da protagonista.

A quasi vent'anni dalla pubblicazione del *Manifesto Agile*, avvenuta nel 2001, e a dieci anni dall'introduzione della metodologia DevOps, con questo paper vogliamo dimostrare che non siamo di fronte a una moda passeggera o a un termine creato dal marketing con il solo scopo di riqualificare un mercato, quello dello sviluppo software, sul quale nel corso degli anni puoi aver avuto motivo di dubitare. L'anzianità di servizio e l'enorme numero di casi applicativi che oggi ne dimostrano la bontà, permettono di affermare che la metodologia DevOps non solo è appropriata ma è necessaria per garantire il successo di un progetto di sviluppo applicativo nei tempi richiesti oggi dal business e, quindi, una maggiore competitività e un più rapido go-to-market.

## Cos'è il DevOps

A un panel dell'ultimo *DevOps Enterprise Summit* di Las Vegas è stata invocata una coppia che non ti aspetti. Perché il Ceo di una delle più importanti aziende specializzate in sviluppo di software aziendale ha sentito il bisogno di citare il duo Lennon&McCartney? Perché le loro caratteristiche e, soprattutto, la loro complementarità messe al servizio della composizione musicale sono un esempio perfetto da riportare in altri contesti. Ebbene, il modello DevOps - termine formato dalla fusione di Developer e Operation - si basa su due caratteristiche, la complementarità e la collaborazione, esattamente le stesse della premiata ditta Lennon&McCartney. Seguendo ancora le analogie del Summit: se la metodologia DevOps può riportare



alla mente il connubio più famoso della musica moderna, d'altra parte deve far dimenticare ciò che è stato l'approccio allo sviluppo fino a qualche anno fa: un silos un po' troppo indipendente e individualista, come Pete Best. Considerando una applicazione come motore di un servizio, l'approccio classico per cui il team di sviluppo lavorava in maniera (troppo) indipendente non funziona più. Come nel gioco del calcio il risultato lo fa solo il collettivo, anche nel modello DevOps si lavora di squadra e i silos scollegati sono evitati per definizione.

DevOps, dunque, si definisce come una metodologia di sviluppo del software che si basa su tre assiomi fondamentali: la comunicazione, la collaborazione e l'integrazione tra la squadra di sviluppatori e quella dei sistemisti, o comunque di tutte le figure aziendali che, in un modo o nell'altro, hanno un ruolo di dipendenza dal servizio applicativo. Ciò vale a maggior ragione in un ambiente cloud, o comunque eterogeneo, in cui è necessario integrare il nuovo servizio all'interno di una piattaforma gestita. L'obiettivo che si pone un approccio DevOps è di sviluppare e gestire in modo più rapido ed efficiente prodotti e servizi software. Come? Attraverso la definizione di una precisa organizzazione delle risorse dedicate al progetto, una maniacale attenzione allo stato di avanzamento dei lavori e un continuo monitoraggio dei servizi in esecuzione.

# Da dove viene il DevOps

Geograficamente il DevOps viene dal Belgio. Fu, infatti, lo sviluppatore belga Patrick Debois che, nel 2009, iniziò a organizzare i primi DevOps Days. Oggi i DevOps Days si svolgono in tutto il mondo praticamente senza soluzione di continuità. In senso lato, invece, il DevOps proviene dalla metodologia Agile, introdotta nel 2001 con il *Manifesto for Agile Software Development* dagli sviluppatori Kent Beck Robert C. Martin, Martin Fowler fra gli altri. Preso atto che i modelli di sviluppo software, primo tra tutti il classico “modello a cascata”, non erano più adeguati alle richieste di business delle aziende, si è scelto di fare tabula rasa.

Quello a cascata è il più tradizionale modello di sviluppo del software applicato in tutto il mondo da almeno 50 anni. Secondo la metodologia classica, si parte da una analisi dei requisiti, si definisce il progetto, si sviluppa, si collauda il software e se ne gestisce la manutenzione, esattamente come si fa per un qualsiasi prodotto industriale.

Ebbene, il modello a cascata oggi fa acqua da tutte le parti per un semplice motivo: la velocità del business non può attendere.

Il modello Agile ha l'obiettivo principale di soddisfare



le richieste del cliente e non semplicemente di onorare un contratto e permette, come dimostrano i numerosi casi applicativi, di abbattere costi e tempi di sviluppo del software, migliorandone la qualità.

Per raggiungere questo risultato il Manifesto definisce quattro principi:

- Le persone e le interazioni tra di loro sono più importanti dei processi e degli strumenti.
- È necessario focalizzarsi sulla qualità del software piuttosto che sulla documentazione rilasciando nuove versioni a intervalli molto frequenti.
- Il cliente è parte attiva della team Agile e deve essere continuamente coinvolto.
- Il team di sviluppo deve essere disposto a modificare in ogni momento le priorità di sviluppo.

Questi assiomi sono anche alla base della metodologia DevOps che, come detto, è una evoluzione dell'approccio Agile. Se quest'ultimo è focalizzato esclusivamente sullo sviluppo, il DevOps permea tutti gli ambienti in cui si propaga un servizio applicativo, come la Rete e il Cloud.

# Perché non puoi fare a meno del DevOps

Nel 1995 Jeff Bezos fonda Amazon e conia il motto "Get Big Fast", motto di cui si appropriarono subito tutte le startup di quegli anni. Non si sbaglia se si immagina che la necessità di rivoluzionare totalmente il metodo di sviluppo del software derivi in qualche modo proprio dalle esigenze delle aziende della New Economy. "Get Big Fast" significa aggredire il mercato di riferimento in modo da crescere il più in fretta possibile. In un momento storico come il 2000 in cui le Dot Com spuntavano come funghi, e a maggior ragione oggi, per sperare di raggiungere il successo della propria startup si doveva essere estremamente rapidi. E, poiché parliamo di un settore di mercato fortemente dipendente dal software, come si poteva pensare di crescere in fretta e aggiungere continuamente nuovi servizi per battere sul tempo il concorrente con un approccio allo sviluppo obsoleto come il "modello a cascata"?

Oggi, per una qualsiasi azienda di un qualsiasi mercato il software è fondamentale e strategico. L'architettura software influenza ogni divisione aziendale e ogni stadio di processo ed è dimostrato che sia un abilitatore del business. Ancora, grazie al digitale, una qualsiasi azienda di un qualsiasi mercato ha l'opportunità di affacciarsi a target sterminato, ma, allo stesso tempo, si trova di fronte una concorrenza agguerrita e molto veloce nel proporre un prodotto o un servizio in più. insomma, il business oggi viaggia a una

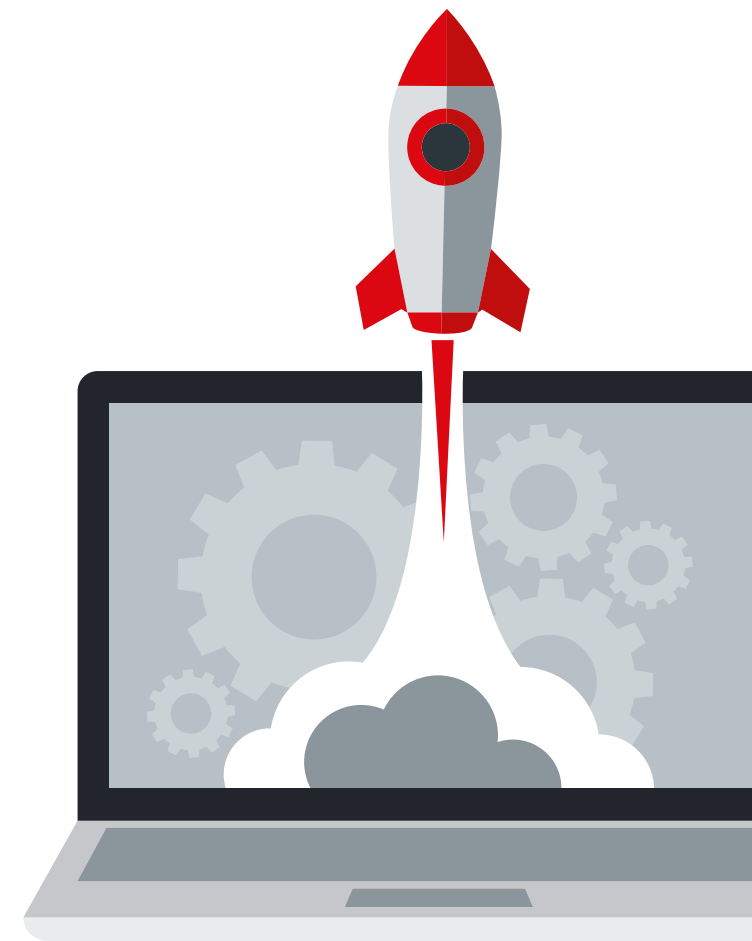
velocità superiore e chi rimane indietro è destinato a soccombere.

Tutte le società di analisi del mercato It, da Gartner a Forrester passando per Idc, sono concordi: per mantenere alta la competitività, oggi nello sviluppo dei propri servizi applicativi un'azienda non può fare a meno di un approccio DevOps. Difficile trovare dati oggettivi del tipo "un'azienda che ha utilizzato l'approccio DevOps è cresciuta dell'x% in un anno" semplicemente perché sarebbe profondamente scorretto imputare a una sola componente la crescita di un'azienda.

Mentre si troveranno diversi casi applicativi e centinaia di motivi che avallano l'ipotesi di partenza: senza DevOps non vai da nessuna parte.

Non vogliamo essere pedanti illustrando le centinaia di motivi per cui dovresti fare un pensierino alla metodologia DevOps, ce ne bastano molti di meno per convincerti:

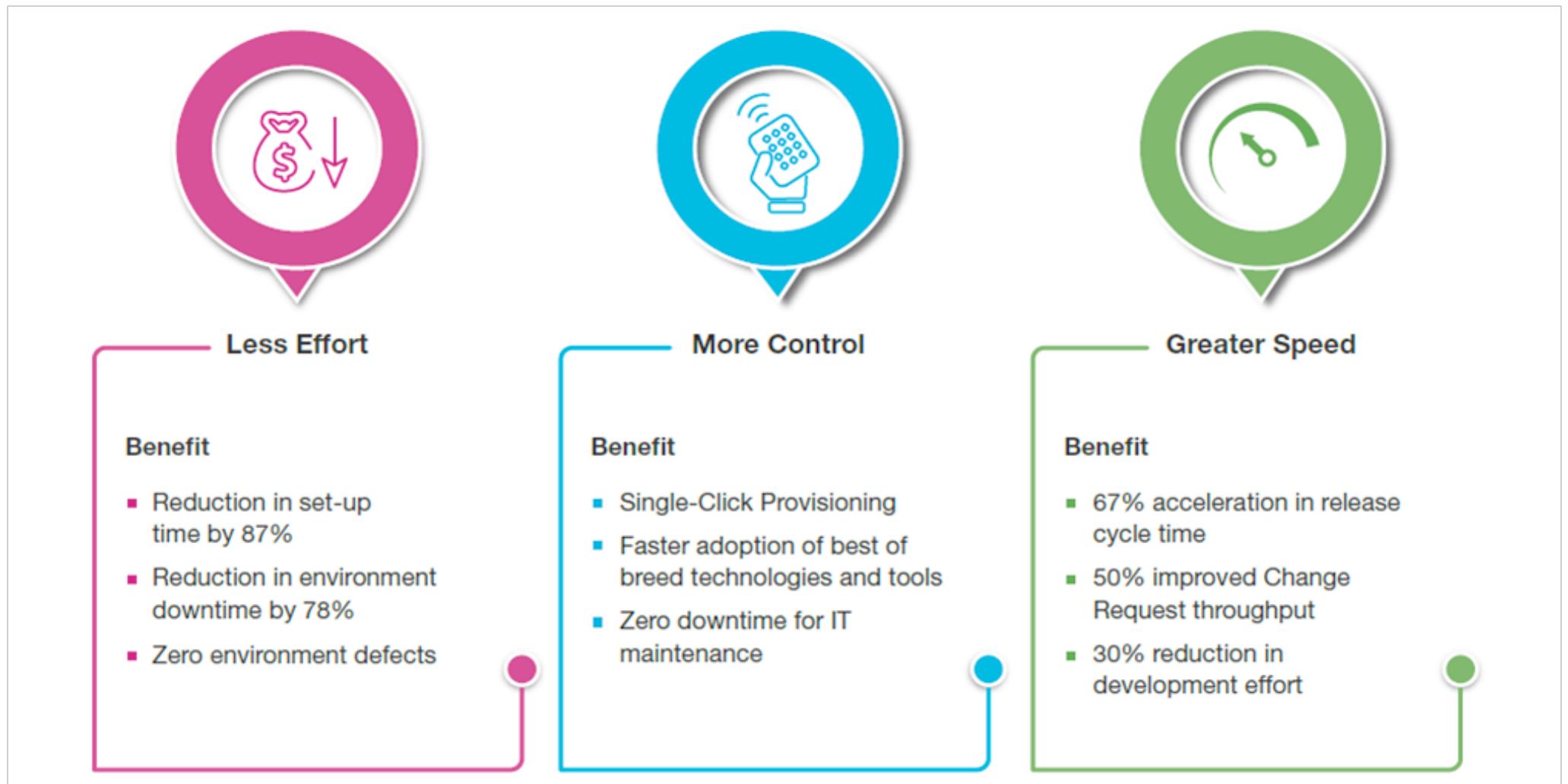
- Forte incremento di velocità nel rilascio delle app, degli aggiornamenti e di nuovi servizi.
- Migliore capacità di localizzazione degli errori e, di conseguenza, più velocità nella risoluzione dei bug e più sicurezza.
- Maggiore partecipazione della struttura It all'ecosistema aziendale con incremento delle sinergie finalizzate all'incremento del business.



**claranet**

hosting | applications | networks

## L'impatto sul business di un approccio DevOps in un progetto di Digital Commerce per una GDO



(Fonte: <https://www.capgemini.com/2016/04/impact-of-digital-agile-and-devops-on-the-delivery-model-of-vendors-and/#>)

# Cosa significa lavorare secondo DevOps

Una volta convinto che il DevOps è ciò che fa per te e per la tua azienda, non ti resta che rimboccarti le maniche e organizzarti per una piccola rivoluzione. Abbiamo detto che l'approccio DevOps si basa sui principi di comunicazione, collaborazione e integrazione principalmente tra due team, ma che alla distanza potrebbe e dovrebbe coinvolgere tutta l'azienda secondo il paradigma più ampio del Business Agile. In particolare, parliamo del team di sviluppo e del gruppo di persone impegnate nelle "operations", ovvero principalmente chi è responsabile del corretto funzionamento dell'applicazione all'interno di una piattaforma It, della sua disponibilità in Rete e dell'integrazione con altri servizi, per esempio via Api o in Cloud. In verità, un organigramma DevOps potrebbe prevedere anche figure non propriamente "informatiche". Ci potrebbe essere bisogno di uno specialista in usabilità dell'interfaccia, per esempio, per controllare in tempo reale l'effetto dei rollout da front end. Oppure un project manager che vigili sulla conformità dell'applicativo al business e alla missione aziendale, o un esperto in compliance che garantisca l'aderenza alle normative.

Che si abbia a disposizione una opportuna struttura It interna o che si decida di affidarsi a un partner esterno – il che è consigliabile almeno nella fase di pianificazione –,

in un progetto DevOps l'organizzazione e la definizione dei ruoli è determinante. Così come è necessaria la scelta e la configurazione di tool software specifici per gestire e controllare lo sviluppo del servizio. Ne esistono una marea e possono assolvere diversi compiti. Tipicamente si tratta di framework per lo sviluppo in grado non solo di enfatizzare e controllare tutti gli aspetti di gestione di progetto ma anche di segnalare in tempo reale criticità o opportunità di miglioramento. Potresti optare per un singolo tool "universale", oppure scegliere una composizione di strumenti che, insieme, ti permettano di gestire il progetto DevOps in tutta la sua complessità. Anche in questo caso, è consigliabile affidarsi a un partner esterno che sappia scegliere i tool più opportuni per lo specifico progetto. Si noti che i framework a supporto prevedono fin da subito la definizione di ruoli e gerarchie delle risorse da impiegare.

Una fondamentale differenza tra la metodologia DevOps e quella classica è che, all'interno del team di sviluppo, si lavora contemporaneamente su moduli distinti da rendere disponibili in continua distribuzione e integrazione. Secondo l'approccio Agile, infatti, è opportuno favorire la formazione di team di sviluppo piccoli, cross-funzionali e auto-organizzati, che garantiscano uno sviluppo iterativo e incrementale, una

## Le 7 figure fondamentali in un team DevOps

1. DevOps Evangelist
2. Release Manager
3. Automation Architect
4. Software Developer/Tester
5. Experience Assurance (XA) Professional
6. Security Engineer
7. Utility Technology Player o DevOps Engineer



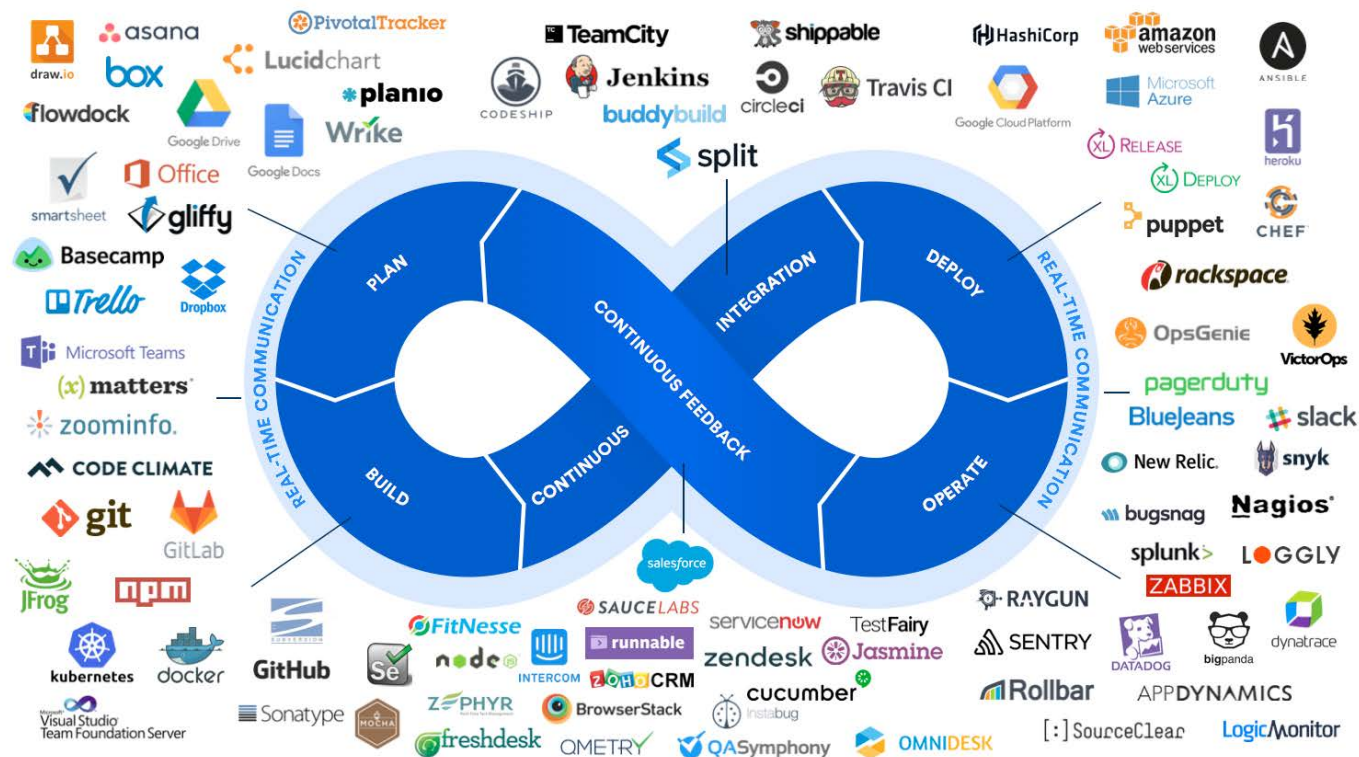
pianificazione adattiva, ovvero capace di modificarsi in divenire.

Attenzione, però, se opti per il supporto di un partner, non puoi pensare di tirarti fuori dal processo. Proprio il coinvolgimento diretto e continuo del cliente – inteso come tutte le divisioni aziendali coinvolte - nel processo di sviluppo è un asset fondamentale per la riuscita del progetto DevOps.

Il risultato deve essere tale per cui appena una componente software è pronta per spiccare il volo, ovvero regge da sola il funzionamento di un servizio, la si rilascia immediatamente, avviando così la macchina del rollout. Da questo momento il progetto non è concluso ma è appena all'inizio. Il team deve essere in grado di rilasciare continui aggiornamenti migliorativi e deve essere pronto a intervenire in tempo reale – e modificare le priorità - in caso di malfunzionamenti, bug di sicurezza, ambiguità nella usability e integrazione di funzionalità aggiuntive.

Rispetto alla produzione industriale, insomma, dove non ci si immaginerebbe mai di mettere sul mercato un semilavorato e, tantomeno, pensare di migliorarlo o integrarlo in divenire, l'approccio DevOps è qualcosa di completamente diverso.

Si comprende, quindi, che la perfetta sincronia della macchina di sviluppo sia alla base di un intervento rapido che permetta di allineare il business alle esigenze del momento.



*Gli strumenti di gestione di un progetto DevOps sono numerosi e soddisfano ogni tipo di esigenza.*

# I rischi di un approccio non corretto

Proprio perché ci troviamo di fronte a una metodologia completamente diversa da ciò a cui siamo stati abituati finora, il primo errore da non fare è sottovalutarla. Un approccio DevOps può funzionare soltanto se si è fermamente convinti del suo modello al punto di sponsorizzarlo il più possibile all'interno delle varie divisioni aziendali a partire dalla struttura It. Prima di una metodologia, il DevOps è una cultura – come ricorda il primo principio del Manifesto Agile e l'acronimo CALM (Culture, Automation, Lean, Monitoring, Sharing) -, come sostengono i suoi fondatori, se non si è pronti per abbracciarla è facile fallire.

Pensiamo a come hai lavorato finora: la divisione It all'interno di una azienda è spesso vista come un costo a cui difficilmente si riesce a riferire un contributo diretto al business aziendale.

Se la mancanza di competenze e di visione dei vertici aziendali può aver contribuito a renderti la vita difficile, è

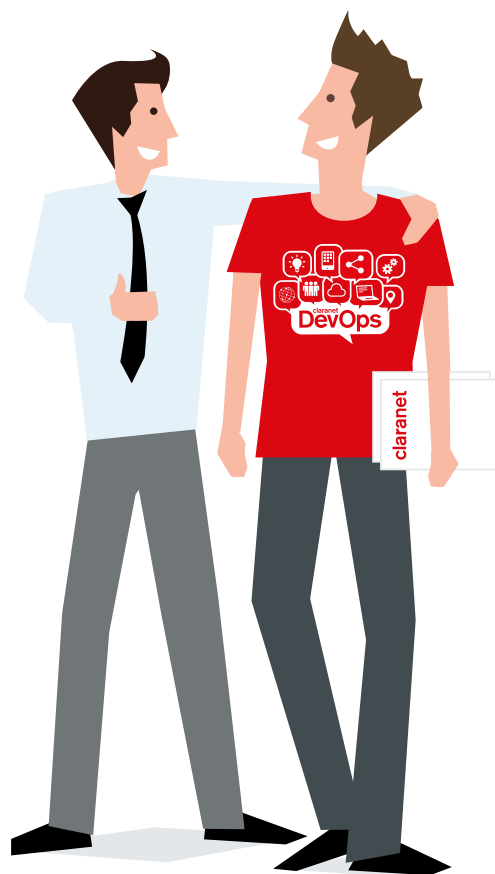


altresì vero che anche tu e il tuo gruppo a volte ci avete messo del vostro.

La filosofia Agile può spingersi addirittura fino a un'estensione di Business Agile, ovvero una cultura collaborativa e interattiva che coinvolga tutta l'azienda in nome di un unico obiettivo: fare business. E, poiché si tratta di una cultura, deve essere propagata con pazienza e passione, perché gli elementi di base sono le persone, non gli strumenti di sviluppo. Gartner, nell'illustrare i motivi per cui un approccio Agile o DevOps fallisce, dichiara: *“Le persone e non i processi sono la principale causa del fallimento di un progetto DevOps”*. È proprio in quest'ambito che l'azienda cliente deve fare lo sforzo maggiore. Non si può lasciare le chiavi dello sviluppo in mano al partner per tornare a progetto concluso ma si deve essere presenti, partecipi, collaborativi e “paladini aziendali” della buona riuscita del progetto e dello stato di avanzamento dei lavori. Solo così la struttura It potrà dimostrare di lavorare per il bene comune: l'azienda.



# Il DevOps per il cloud e poi?



Ma, ti chiederai, abbiamo veramente bisogno di un approccio DevOps? Se non ti basta l'opportunità di presentarsi al mercato in maniera più competitiva, più reattiva e con servizi più efficaci, funzionali e, non ultimo, sicuri, ci aggiungiamo ancora qualcosa. Il ricorso a competenze miste e diverse dai puri manovali del codice lo richiede espressamente il mercato. E non è una frase fatta. Pensa solo a come si è evoluta la tua infrastruttura It, dov'è installato adesso l'applicativo che usano i dipendenti della tua azienda? Molto probabilmente hai già scelto che non sia più collocabile in un preciso luogo fisico. Probabilmente utilizzi architetture di cloud ibride per cui non ha più senso preoccuparsi di dove siano archiviati i dati aziendali. E, probabilmente, hai accettato di utilizzare servizi e applicazioni remoti, per esempio forniti dal tuo cloud provider, per esempio Amazon Web Services. Ciò che hai raggiunto è, insomma, un buon controllo sul software indipendentemente da dove sia installato, ti sei tolto qualche grattacapo e, soprattutto, hai usato bene il tuo budget. Ebbene, per poter configurare quel software, per poterlo modificare, per integrare nuove funzionalità, devi ricorrere a figure professionali con competenze diverse dal puro sviluppo, i generici "operatori" racchiusi nel termine Operations.

Proprio l'attitudine a un'apertura generalizzata, in primis a nuove risorse professionali che non siano solo puri fanatici del codice, ti fa comprendere che il DevOps può far del bene anche in altri reparti aziendali. Non abbiamo la sfera di cristallo, ma immaginiamo che

un'azienda che ha adottato un modello di Business Agile, osi ancora di più e, per esempio, decida di gestire la propria linea di produzione attraverso applicativi e servizi DevOps. In questo modo, per esempio, il miraggio di una produzione in grado di modificare e aggiornare il prodotto con tempi di reazione mai visti prima, improvvisamente ti apparirà concreto. Bello, no?

## Chi è Claranet

Fondata nel 1996, Claranet è un Managed Service Provider specializzato in outsourcing di applicazioni critiche. Con 1.800 dipendenti in 8 paesi, Claranet è diventata uno dei principali operatori di servizi gestiti in Europa (360 milioni di euro di fatturato) e gestisce soluzioni per oltre 6.500 clienti in tutti i settori aziendali, tra cui MyMajorCompany, Energisme e Airbus. Claranet riunisce le migliori competenze, tecnologie e procedure per fornire servizi gestiti flessibili, sicuri e convenienti in grado di supportare le prestazioni e le applicazioni di rete. Con Claranet le aziende possono concentrarsi sul proprio core business dimenticando la gestione del loro IT. Per maggiori informazioni: [www.claranet.it](http://www.claranet.it)