

[Home](#) > [Architecting](#)

FinOps su AWS: governare i costi del Cloud prima che siano loro a governare noi

2 Dicembre 2025 - 6 min. read

FinOps

Introduzione

Che cosa significa FinOps? O meglio, che cos'è?

In questi ultimi anni si è sentito parlare sempre di più di concetti come ottimizzazione dei costi e analisi della bolletta, ma quanti di voi sono mai andati oltre, pensando non solo all'azione ma al significato che c'è dietro e al come applicare al meglio queste pratiche?

Se siete una di quelle persone che ha sempre messo da parte questo argomento, beh questo articolo è specificatamente per voi. Se, invece, conoscete già l'argomento... continuate a leggere comunque: ci sono alcune chicche dalla nostra esperienza da non perdere!

Il Framework FinOps

Come tutti gli standard che si rispettino, anche il Framework FinOps è stato ben documentato e definito; si tratta infatti di un approccio iterativo, suddiviso in 3 differenti momenti:



Inform

Dovendo prendere decisioni proiettate sul futuro, dobbiamo prima capire bene il presente e il passato. Infatti, in questa fase ci si concentrerà sulla raccolta, l'analisi e la standardizzazione di informazioni.

Come disse un saggio: "I dati raccontano una storia, ma devi quantomeno saperli leggere..."

Un grande obiettivo che dobbiamo prefissarci, oltre alla raccolta di dati e alla loro standardizzazione, è il come renderli confrontabili tra diversi gruppi di lavoro o progetti. I KPI, quindi, risultano tanto importanti quanto il saper leggere i dati.

Immaginiamoci che la raccolta dati sia il decollo di un aereo e che l'analisi sia il volo. Dovremmo riuscire comunque ad atterrare sani e salvi, altrimenti il valore di quello che abbiamo fatto si perderebbe!

Optimize

A questo punto, abbiamo finalmente delle idee chiare su che cosa è superfluo o su quali componenti dobbiamo abbattere i costi. Non ci resta che concretizzare.

Il Cloud ci rende questo processo più semplice: alcune opzioni, come le 'Reserved Instances', 'Saving Plans' o 'Committed Use Discounts' contribuiscono a tagliare i costi in bolletta, ma riguardano la bonifica di una situazione già in atto; il vero processo di FinOps deve partire sin da subito, dalla nascita di un progetto.

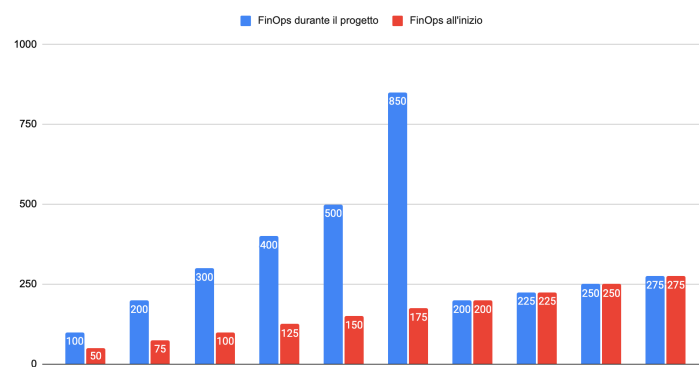
Una delle difficoltà maggiori di questa fase è il doversi interfacciare con diverse Persone o Team all'interno della stessa azienda o, addirittura, di differenti aziende. Questo potrebbe comportare dei momenti di tensione e di competizione, ma ricordiamoci sempre che alla fine stiamo remando tutti verso la stessa direzione.

Operate

In questo punto del percorso abbiamo preso delle decisioni informate basate sui dati e ridotto la bolletta. Abbiamo quindi risolto il problema, no?

Beh, non proprio.

Confrontiamo due grafici:



Nel primo esempio (serie in blu), non siamo partiti con un approccio FinOps sin dall'inizio, incombendo in costi evitabili, per poi raggiungere un picco insostenibile per il progetto e dover rifattorizzare il tutto.

Il refactoring ha senz'altro abbassato drasticamente i costi, ma la bolla iniziale di costo ha comunque aumentato di molto la spesa, portando il totale a 3300\$.

Prendiamo invece il secondo caso (serie rossa), in cui sin da subito si ha avuto un'ottica improntata alle Best Practices FinOps, mantenendo un costo ottimizzato sin da subito. Non dovendo poi effettuare dei grossi cambiamenti durante il ciclo di vita del progetto, il costo totale è risultato di 1625\$.

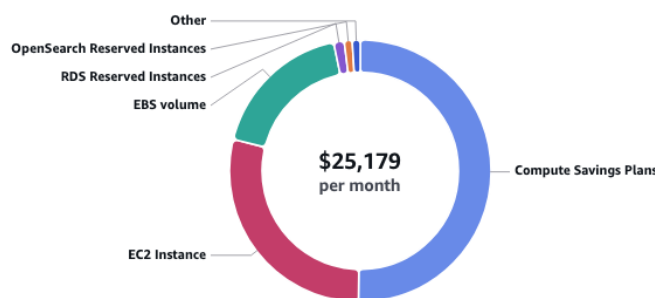
In questi casi risulta essere molto utile partire con un TCO (Total Cost of Ownership), il quale ci permette di stimare sin da subito quanto costerà la nostra implementazione, andando eventualmente a efficientare le aree che ce lo permettono, abbattendo i costi di sviluppo e di mantenimento del progetto.

Risulta dunque fondamentale avere e coltivare una cultura attorno alle pratiche FinOps, così da garantire sempre dei progetti sostenibili e efficienti dal punto di vista economico.

Come possiamo applicarlo su AWS?

Abbiamo parlato della teoria. Ora cerchiamo di capire come poterla mettere in pratica.

Partiamo dai servizi che ci possono aiutare a comprendere la situazione dei nostri account, ad esempio, Cost Optimization Hub che offre una visione sui costi generale, ma completa:



Qui, possiamo osservare quali sono le opzioni disponibili per iniziare ad effettuare delle ottimizzazioni.

Cost Optimization Hub non svolge operazioni complesse, ma controlla semplicemente quali risorse non sono usate, come EBS non in uso, e se possiamo sfruttare Saving Plan o Reserved Instance: semplice ma efficace.

Il servizio sfrutta un insieme di altri dati, come ad esempio Compute Optimizer.

Stiamo però toccando solamente la punta dell'iceberg.

Ad esempio, potremmo sottoscrivere dei Saving Plan ottimizzando così il costo, ma non dovremmo fermarci una volta effettuata questa prima bonifica.

Al posto di utilizzare passivamente i servizi AWS, dunque, dovremmo invece cercare di porci delle domande mirate e ragionate per esempio:

- Possiamo modernizzare questa applicazione?
- Ci sono soluzioni gestite che ci permettono di abbassare i costi?

- Possiamo effettuare del decoupling del monolite e gestirne i picchi dinamicamente con delle politiche di scaling?

Così facendo, ci assicuriamo di non accettare senza consapevolezza l'infrastruttura attuale, ma di comprendere e definire il perché di queste decisioni e azioni.

Facciamo un esempio concreto: se avessi un centesimo per ogni volta che ho migrato un web server monolitico utilizzato per fare hosting di un sito statico da on-premise verso AWS, ne avrei due. Il che non è molto, ma è strano che sia successo due volte.

Se al posto di migrare direttamente il sito statico avessimo affrontato il contesto e modernizzato la soluzione sfruttando Cloudfront e S3, avremmo potuto fornire un servizio migliore, ad una frazione del costo di una EC2 dedicata.

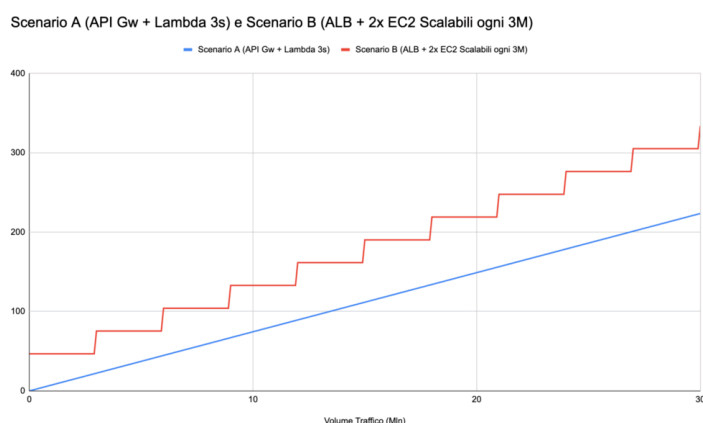
Ovviamente non sempre è possibile effettuare queste operazioni per limiti tecnici o per situazioni non documentate; in questo caso si può però ragionare sul costo del mantenimento rispetto al costo di ammodernamento!

Questa strategia non solo si può, ma si deve applicare in continuazione, proprio per come il cloud provider AWS ha concepito il modello di *Pricing*.

Facciamo un esempio reale, senza entrare troppo nel dettaglio della casistica, ma concentrandoci sulla struttura del costo:

Nel grafico qui sotto abbiamo due casistiche:

- Un API Gateway che espone delle API ed inoltra le richieste a delle Lambda, il cui tempo medio di risposta è 3 secondi per ogni chiamata
- Un autoscaling group composto da due istanze EC2 che espongono API tramite un ALB. Abbiamo ipotizzato la creazione di due istanze per ogni 5 Milioni di richieste



In questo caso possiamo vedere come alcuni servizi gestiti possano avere le stesse funzionalità, o in alcuni casi anche di più, ad un costo minore, essendo ottimizzati proprio per quel caso d'uso.

In questo specifico caso, dovremmo però contare anche un delta di costo per il refactor dell'applicativo per renderlo utilizzabile da API GW e Lambda. Bisognerà quindi valutare caso per caso qual è la soluzione migliore e più efficiente, sia economicamente che tecnologicamente.

In alcuni casi, seppure possa sembrare controindicato, si potrebbero presentare delle infrastrutture serverless che costano più di alcune serverfull. Non in questo caso, ma pensate se la Lambda al posto di essere eseguita in 3 secondi ne impiegasse 10; sarebbe più che triplicato il costo, spostando di molto in alto la 'Linea Blu' del grafico.

Questo articolo è un'introduzione allo standard FinOps, ma anche al nostro approccio a riguardo. È anche il nostro punto di vista su ciò che riteniamo possa dare valore al cliente e su quali siano le domande-chiave che ci possono spingere verso un migliore approccio al cloud.

Ma non è tutto qui!

Presto torneremo con una seconda parte dove ci concentreremo sulle domande che siamo posti in questi paragrafi in relazione all'applicazione dell'approccio FinOps e di come tutto ciò abbia permesso di ottimizzare, e in certi casi anche di modernizzare, alcuni *Workload* di clienti.

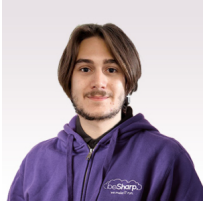
Speriamo che, da ora in poi, il vostro punto di vista su cosa sia realmente FinOps possa essere arricchito. Fatecelo sapere!

A presto con la seconda parte.

About Proud2beCloud

Proud2beCloud è il blog di **beSharp**, APN Premier Consulting Partner italiano esperto nella progettazione, implementazione e gestione di infrastrutture Cloud complesse e servizi AWS avanzati. Prima di essere scrittori, siamo Solutions Architect che, dal 2007, lavorano quotidianamente con i servizi AWS. Siamo innovatori alla costante ricerca della soluzione più all'avanguardia per noi e per i nostri clienti. Su Proud2beCloud

condividiamo regolarmente i nostri migliori spunti con chi come noi, per lavoro o per passione, lavora con il Cloud di AWS. Partecipa alla discussione!



Stefano Salvaneschi

DevOps Engineer @ beSharp. Sono la classica persona che non accetta che una domanda possa rimanere senza risposta. Sia quando ne faccio (molto spesso!), sia quando ne ricevo. Appassionato di IT fin da quando ero un bambino, sono da poco approdato nel mondo del Cloud. Nel tempo libero amo giocare ai videogames, a dungeons & dragons e ad altri giochi di ruolo tipicamente "nerd"... il tutto accompagnato da una buona birra!

Copyright © 2011-2025 by beSharp spa - P.IVA IT02415160189