

Ottenere observability nelle applicazioni cloud-native



Contenuto



01 →	Introduzione	06 →	I rischi della mancata adozione dell'observability in ambienti cloud-native
02 →	Tecnologie basate su cloud e cloud-native a confronto	07 →	Benefici dell'observability per applicazioni cloud-native
03 →	L'observability in ambienti cloud-native	08 →	Introduzione all'observability aziendale
04 →	Sfide di observability	09 →	Informazioni su IBM® Instana
05 →	Caratteristiche critiche dell'observability per ambienti cloud-native		

Introduzione

L'adozione diffusa delle diverse tecnologie cloud-native ha generato cambiamenti significativi nello sviluppo delle applicazioni, nella consegna e nelle operazioni, portando a un nuovo landscape competitivo. Le organizzazioni più veloci stanno approfittando di questo cambiamento per superare i loro corrispettivi più grandi e più lenti.



Le variazioni allo stack tecnologico delle applicazioni sono state rapide. Gli ambienti cloud, pubblici e privati si sono espansi, i server virtuali si sono evoluti in container e Kubernetes è diventato lo standard di orchestrazione. L'adozione dei container è esplosa ed si è evoluta in workload serverless negli ambienti serverless, mentre i team di sviluppo sono diventati più agili. Per via della confusione tra software e infrastruttura, sono emerse due costanti concorrenti:

- Il monitoraggio delle prestazioni delle applicazioni (APM) è diventato un importante coadiutore al successo aziendale.
- Era difficile ottenere visibilità sulle applicazioni moderne utilizzando le metodologie tradizionali di APM.

In questo modo, il concetto di observability, nelle diverse forme, è emerso mentre le organizzazioni lottavano con la necessità di accelerare lo sviluppo e la consegna, mantenendo al contempo la visibilità che i team di operazioni IT si aspettavano dagli strumenti APM. L'observability, però, comporta delle sfide, tra cui molte delle stesse che affrontano gli strumenti di monitoraggio tradizionali. Le organizzazioni che si occupano della distribuzione di applicazioni in modo rapido e flessibile, come Dev, Ops, DevOps e SRE, desiderano utilizzare l'intera gamma di tecnologie cloud-native. Per farlo, hanno bisogno di un approccio completo alla observability, che utilizzi al meglio le best practice della visibilità guidata dagli sviluppatori, e delle soluzioni APM moderne.

Basata sul cloud vs le tecnologie cloud-native

La tecnologia cloud è stata una vera svolta in termini di sviluppo delle applicazioni. Gli ambienti cloud e multicloud offrono maggiore potenza e ridondanza, il che richiede la modernizzazione delle applicazioni. Mentre le applicazioni basate su cloud sono state adattate per le operazioni nel cloud, le applicazioni cloud-native sono state sviluppate specificamente per la distribuzione su cloud. Tuttavia, a quale cloud ci riferiamo? Man mano che le aziende optano per ambienti multicloud, la complessità aumenta. Gli hybrid cloud, con i quali si intendono una combinazione di cloud privati e pubblici, aumentano ulteriormente questa complessità. Quando aggiungiamo un livello di hosting on-premise al mix, gli stack

moderni di tecnologia possono esercitare ulteriore pressione sui team di sviluppo, rendendo difficile la consegna tempestiva delle applicazioni.

Il monitoraggio delle prestazioni delle applicazioni cloud-native può rivelarsi complesso, poiché gli strumenti tradizionali di APM e observability potrebbero avere difficoltà nell'esaminare gli elementi cloud-native, come i container, con la stessa facilità con cui esaminano altri stack applicativi. Al contrario, le piattaforme di observability cloud-native tendono a tracciare tutte le metriche, persino le tracce degli eventi e i log nel full stack per offrire una visibilità più ampia.



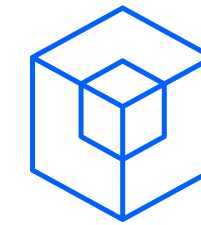
Le applicazioni basate su cloud sono state adattate per le operazioni nel cloud.



Le applicazioni cloud-native sono state sviluppate specificamente per il cloud.

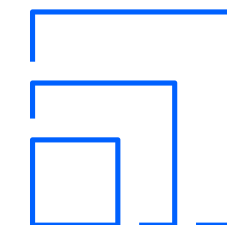
L'observability in ambienti cloud-native

Le tecnologie che compongono i moderni stack tecnologici cloud-native presentano alcuni ostacoli specifici per gli strumenti di monitoraggio tradizionali:



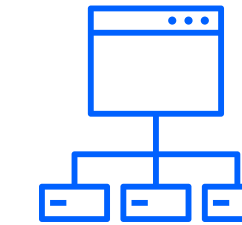
Mancanza di retrocompatibilità

I container sono progettati appositamente per contenere solo i servizi necessari in modo da mantenerli leggeri, il che significa che i motori di strumentazione più vecchi potrebbero non avere accesso ai servizi necessari per funzionare.



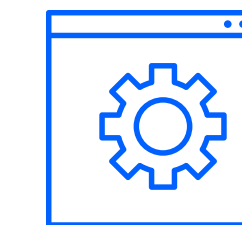
Eliminazione serverless

Le architetture serverless portano l'ideale dell'eliminazione dei servizi a un livello ancora più elevato, trasferendo i server dal controllo locale alla piattaforma cloud, con conseguenti problemi di disponibilità dei servizi.



Correlazione delle informazioni di orchestrazione

Kubernetes offusca i confini tra software e infrastruttura al punto che gli strumenti esistenti potrebbero dover usare più agenti solo per estrarre dati di base, eliminando così la capacità di correlare direttamente le informazioni.



Accelerazione dello sviluppo e dei tassi di distribuzione

Poiché l'ambiente applicativo è soggetto a continui cambiamenti a causa della natura effimera dei container, può essere difficile per gli strumenti di monitoraggio tradizionali raggiungere il loro consueto livello di visibilità.

Poiché gli strumenti APM tradizionali faticavano a soddisfare i requisiti degli ambienti cloud-native, sono emerse alternative per l'observability delle applicazioni grazie ad altri fattori::

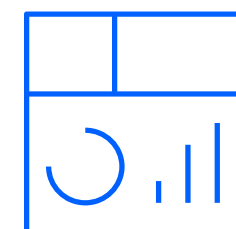
- Gli sviluppatori sono stati coinvolti maggiormente nella gestione dei workload di produzione e nella garanzia delle prestazioni delle applicazioni.
- Gli stack tecnologici cloud-native hanno consentito uno sviluppo accelerato e pipeline applicative, il che significa che più componenti potrebbero operare con lacune di visibilità rispetto agli strumenti esistenti.

L'observability fornisce indicatori esterni dello stato di salute e dello stato delle applicazioni all'interno di elementi cloud-native quali container, microservizi, strumenti di orchestrazione come Kubernetes e ambienti serverless dal punto di vista delle prestazioni.



Sfide di observability

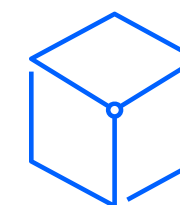
Dalle "black box" (meccanismi interni nascosti di un prodotto o programma) alle implementazioni di breve durata, lo stack cloud-native crea numerose sfide per gli strumenti di observability e APM:



Gli strumenti di monitoraggio tradizionali non riescono a gestire ambienti di microservizi distribuiti.

Gli ambienti per applicazioni basate su container tendono a essere microservizi dinamici e distribuiti, implementati tra cluster di server che sono anch'essi dinamici. Un'applicazione completa potrebbe attraversare diversi tipi di infrastrutture, piattaforme, linguaggi e tecnologie.

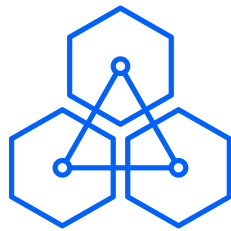
Gli strumenti di monitoraggio delle Infrastrutture e delle applicazioni sono stati progettati per applicazioni monolitiche mappate su singoli server statici e tendono a concentrarsi sulla visibilità a livello di linguaggio. Questi approcci tradizionali di monitoraggio possono fallire di fronte ai microservizi distribuiti su un ampio cluster di server composti da più linguaggi, componenti middleware e sistemi database.



I container ospitano una grande varietà di workload

I container possono ospitare vari tipi di workload, che vanno da applicazioni monolitiche come Apache HTTP server a macchine virtuali e database Java, creando sfide complesse di monitoraggio. Il data storage persistente in outsourcing su server permanenti in un ambiente ibrido complica ulteriormente le cose. Con una tale variabilità del workload negli ambienti containerizzati, gli strumenti di monitoraggio devono supportare una vasta gamma di tecnologie, architetture e configurazioni, rendendo questo compito piuttosto arduo.

Gli strumenti di monitoraggio tradizionali possono essere difficili da configurare per i container, poiché ogni container è diverso ed è difficile determinare in anticipo le soglie da impostare per ciascuno. Inoltre, gli strumenti di monitoraggio devono essere automatizzati per gestire tipi di workload imprevedibili, poiché la configurazione manuale e l'applicazione delle politiche di monitoraggio non sono fattibili.



I container richiedono un monitoraggio elevato

Le piattaforme di container includono tipicamente solo funzionalità di monitoraggio di base, ma lo strumento statistico è insufficiente per monitorare ambienti di produzione su larga scala. Ciò distingue i container dalle altre tecnologie di infrastrutture applicative.

Allo stesso modo, i sistemi operativi forniscono dati significativi sulle prestazioni del sistema, ma non includono dati a livello di applicazione. Anche piattaforme avanzate di macchine virtuali come VMware, che include strumenti di monitoraggio relativamente sofisticati, non forniscono dati a livello di applicazione o tracciamento distribuito.



Kubernetes non è uno strumento di monitoraggio delle prestazioni
Gli orchestratori di container come Kubernetes sono ottimi per organizzare, fornire e gestire l'implementazione dei propri ambienti di produzione containerizzati e relative applicazioni.

Ma gli orchestratori non sono progettati per il monitoraggio e, sebbene abbiano un enorme focus su container e risorse host, la gestione dell'orchestrazione non include alcun aspetto dell'esperienza utente o delle prestazioni applicative.

Inoltre, orchestratori di container possono gestire solo ciò di cui sono a conoscenza, cioè l'infrastruttura e i servizi che girano nei container. In un ambiente ibrido, gli orchestratori di container, per definizione, non monitorano l'utilizzo delle di nessun altra risorsa. Questo può lasciare gran parte dell'ambiente a rischio di problemi di prestazioni.



Caratteristiche critiche dell'observability per le applicazioni cloud-native ambienti

La visibilità è il cuore pulsante delle piattaforme di observability cloud-native. La capacità di monitorare i container dall'interno è una caratteristica critica per architetture dinamiche basate su microservizi, in cui il tracciamento e le mappe di dipendenza possono essere una sfida.

Anche con la natura effimera e talvolta incompleta dei container, l'observability cloud-native fornisce insight attuabili per trasformare i team da osservatori passivi in partecipanti attivi capaci di risolvere e persino prevenire problemi. Gli agenti esperti di microservizi containerizzati possono mappare le risorse in un'architettura IT, anche quando queste risorse sono a malapena collegate e in costante cambiamento.

Altre caratteristiche critiche includono:

Tutti i dati

DevOps deve avere accesso a un'analisi completa per comprendere esattamente come, quando e dove sussistono problemi. L'unico modo per raggiungere questo obiettivo è catturare una traccia distribuita di ogni richiesta, nessuna lacuna, nessun campionamento e nessun proxy. Inoltre, ogni parte di infrastruttura, piattaforma e linea di codice implementata nell'ambiente containerizzato deve supportare il tracciamento.

Distribuzione cloud-native

Gli strumenti di observability cloud-native si integrano perfettamente in ambienti applicativi cloud-native, processi di distribuzione e strumentazione completamente automatizzati.

Analisi della causa principale

Indipendentemente dalla complessità del tuo ambiente, i team DevOps devono identificare e risolvere le problematiche delle prestazioni il più rapidamente possibile. Una soluzione di observability deve essere in grado di indicarti automaticamente dove e perché le applicazioni distribuite si interrompono.

I rischi della mancata adozione dell'observability per gli ambienti cloud-native



Per alcune organizzazioni, implementare l'observability per le applicazioni cloud-native potrebbe non essere una priorità, forse perché la loro soluzione APM non presenta un numero allarmante di errori o perché i loro team avrebbero dovuto imparare una nuova piattaforma.

Ma se la tua soluzione di monitoraggio tradizionale non rileva errori, latenza delle transazioni e opportunità di ottimizzazione, potresti non renderti conto che hai bisogno di una piattaforma di observability aziendale. Il ritardo comporta dei rischi:

- Un errore non rilevato potrebbe causare un peggioramento del servizio rivolto ai clienti o un'interruzione.
- I ritardi, dal momento del rilevamento dell'incidente al momento della scoperta da parte di un essere umano, potrebbero fare la differenza tra la prevenzione degli incidenti e la gestione degli incidenti di grave entità.
- Il tuo strumento di monitoraggio potrebbe individuare un problema, ma non fornire abbastanza informazioni contestuali per farti intervenire.

Tutti questi scenari potrebbero potenzialmente causare danni alla reputazione, perdita di ricavi dovuti a tempo di inattività e ripercussioni sull'esperienza del cliente.

Benefici dell'observability per le applicazioni cloud-native

Naturalmente, il principale vantaggio di una soluzione di monitoraggio più completa sta nell'individuare problemi e prevenire incidenti. Grazie all'observability, i sistemi trasparenti e osservabili consentono agli ingegneri e agli SRE di liberare il tempo, precedentemente necessario per il monitoraggio continuo, e concentrarsi su attività a più alto valore aggiunto, come la creazione di nuove caratteristiche di prodotto per i clienti.

Processi di integrazione/distribuzione continua (CI/CD) più rapidi, meno sprechi di lavoro, meno errori e clienti soddisfatti consentono di risparmiare tempo e denaro. Ma con IBM Instana puoi fare molto di più:

Riduci il tempo medio di riparazione (MTTR)

La correlazione degli eventi può fare la differenza tra valutare un incidente per ore e risolvere un problema imminente prima ancora che si verifichi.

Shift-left

Quanto prima ingegneri e sviluppatori vengono a conoscenza di un problema, tanto più possono essere proattivi e meno reattivi, e tanto minore sarà l'impatto di ogni problema.

Riduci l'impatto CI/CD

L'observability rende più facile il tracciamento del successo di un rilascio canary o di una distribuzione blu/verde in corso. Una volta rilasciato il codice non riuscito in produzione, il danno è fatto.

Sistemi più sani

L'output naturale di uno strumento di observability in un ambiente cloud-native con DevOps significa sistemi con meno errori e processi più efficienti.

Clienti più soddisfatti

Come cliente, per quanto tempo continueresti a pagare un'applicazione o un servizio che presenta errori, risponde lentamente, comporta tempi di inattività e minaccia la tua attività?

Introduzione all'observability aziendale



Sebbene l'observability possa fornire i segnali necessari per monitorare le prestazioni delle applicazioni, per ottenere una comprensione davvero completa dell'intero sistema è preferibile ricorrere all'observability aziendale.

Con l'observability aziendale, puoi fare molto più che monitorare i singoli sistemi. Puoi contestualizzare i dati relativi a tali sistemi e correlare le interazioni tra applicazioni e sistemi in tutto il tuo ambiente IT.

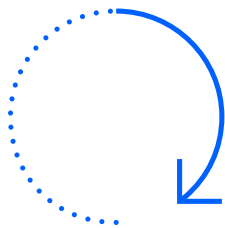
Le basi dell'observability aziendale sono automazione, contesto, azione intelligente, correzione e facilità d'uso.

Più nello specifico, l'observability aziendale si basa su diverse pratiche chiave e principi:

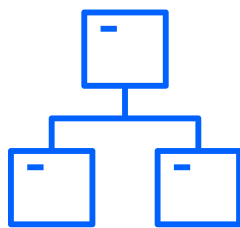
- **Ottimizzazione sistematica**
L'observability aziendale non si concentra sulla gestione dello stato di salute e delle prestazioni delle singole applicazioni o sistemi, ma sull'ottimizzazione dell'intero ambiente IT. Questo significa mappare e contestualizzare tutte le risorse all'interno dell'architettura, anche se sono in costante cambiamento o debolmente accoppiate.
- **Contestualizzazione completa**
Ogni unità di dati di observability deve essere fornita con un contesto completo. I team non possono affidarsi al campionamento e alle supposizioni, hanno bisogno di tracciabilità e contestualizzazione end-to-end.

- **Distribuzione cloud-native**
L'observability aziendale deve essere in grado di integrarsi senza soluzione di continuità negli ambienti applicativi cloud-native che supportano. I processi di distribuzione e strumentazione sono completamente automatizzati.
- **Supporto completo per la data ingestion**
Gli ambienti applicativi aziendali moderni espongono i dati in vari modi: come output standard o log convenzionali, oppure tramite API di monitoraggio open source. Gli strumenti di observability aziendale devono supportare tutti i formati di dati per inserire e contestualizzare ogni fonte di dati.

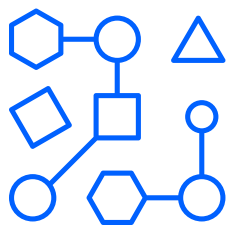
- **Observability attraverso la pipeline**
I team devono essere in grado di monitorare e contestualizzare il comportamento delle applicazioni dall'inizio della pipeline CI/CD fino alla distribuzione. Non dovresti aspettare che una nuova versione dell'applicazione sia in produzione per poter capire come interagisce con altri sistemi e ottimizzarne il comportamento.



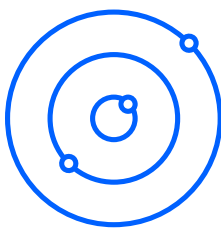
Automazione
Quando viene implementato un nuovo codice o vengono apportate modifiche al sistema, l'observability aziendale rileva automaticamente e continuamente le modifiche e fornisce un feedback immediato.



Contesto
L'observability aziendale spiega come ogni componente e servizio applicativo si colleghi a tutti gli altri componenti e servizi per ottimizzare le prestazioni e la disponibilità di risorse.



Azioni intelligenti
Quando avvengono cambiamenti, l'observability aziendale fornisce proattivamente analisi approfondite con il contesto e suggerisce passaggi per ottimizzare il sistema.



Facilità d'uso
Man mano che le organizzazioni implementano processi di sviluppo agile, aumentano la frequenza degli aggiornamenti delle loro applicazioni e riempiono le loro pipeline CI/CD, con un numero sempre maggiore di stakeholder che richiede informazioni fruibili.

Informazioni su IBM Instana



IBM Instana fornisce una [piattaforma di observability aziendale](#) con [funzionalità di monitoraggio automatizzato delle prestazioni applicative](#) alle aziende che gestiscono applicazioni complesse, moderne e cloud-native, indipendenti dalla collocazione, on-premise o in cloud pubblici e privati, su dispositivi mobili o IBM zSystems.

Controlla le moderne applicazioni ibride con la scoperta basata su AI di IBM Instana delle dipendenze contestuali profonde all'interno delle applicazioni ibride. IBM Instana offre anche visibilità sulle pipeline di sviluppo, per facilitare l'automazione DevOps a loop chiuso.

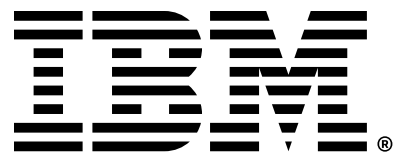
Queste funzionalità forniscono feedback attuabili utili ai clienti per ottimizzare le prestazioni delle proprie applicazioni, favorire l'innovazione e mitigare i rischi, aiutando DevOps a migliorare l'efficienza e a valorizzare le pipeline di consegna del software, raggiungendo al contempo gli obiettivi di servizio e di business.

[Esplora IBM Instana](#)



[Prova gratuita di IBM Instana](#)





© Copyright IBM Corporation 2023

IBM Corporation
New Orchard Road
Armonk, NY 10504

Prodotto negli Stati Uniti d’America
maggio 2023

IBM, il logo IBM, IBM Instana e zSystems sono marchi o marchi registrati di International Business Machines Corporation, negli Stati Uniti e/o in altri Paesi. Altri nomi di prodotti e servizi potrebbero essere marchi di IBM o di altre società. Un elenco aggiornato dei marchi IBM è disponibile all'indirizzo ibm.com/it-it/trademark.

Le informazioni contenute nel presente documento sono aggiornate alla data della prima pubblicazione e possono essere modificate da IBM senza preavviso. Non tutte le offerte sono disponibili in ogni paese in cui opera IBM.

La valutazione e la verifica del funzionamento di qualsiasi altro prodotto o programma con prodotti e programmi IBM sono responsabilità dell’utente. LE INFORMAZIONI RIPORTATE NEL PRESENTE DOCUMENTO SONO DA CONSIDERARSI "NELLO STATO IN CUI SI TROVANO" SENZA GARANZIE, ESPLICITE O IMPLICITE, IVI INCLUSE GARANZIE DI COMMERCIALIZZABILITÀ, DI IDONEITÀ A UN PARTICOLARE SCOPO E GARANZIE O CONDIZIONI DI NON VIOLAZIONE. I prodotti IBM sono coperti da garanzia secondo i termini e condizioni dei contratti sulla base dei quali vengono forniti.