

WHITE PAPER

UNA PIATTAFORMA PER LA MODERNIZZAZIONE DELLE APPLICAZIONI JAVA

Progettazione di carichi di lavoro per gli ecosistemi cloud e moderni

Il 72% dei costi IT
riguarda progetti di
manutenzione.²

Il 67% dei
CEO integrerà la
trasformazione digitale
come elemento chiave
della propria strategia
di business.¹

Il 59% dei CIO
teme che le proprie
competenze attuali in
ambito IT non siano
più sufficienti.²⁰

Il 40% delle aziende
intende rinnovare le
proprie infrastrutture
IT entro i prossimi
3 anni.¹⁸

RIEPILOGO ESECUTIVO

Le organizzazioni sono davanti a un bivio strategico. Secondo gli analisti IDC,¹ due terzi dei CEO incentreranno le strategie aziendali su iniziative di trasformazione digitale. L'obiettivo della trasformazione digitale è solo in parte quello di rendere le attuali funzioni più efficienti. In un panorama più ampio, invece, lo scopo è applicare nuove strategie che consentano di raggiungere i nuovi obiettivi e di sfruttare meglio i dati esistenti.

Tuttavia, il 72% degli attuali costi dell'IT² - e dell'impiego delle risorse IT - è rivolto alla manutenzione dei sistemi esistenti. Ciò genera un punto di attrito tra le priorità attuali e quelle future.

Secondo IDC,³ le applicazioni middle-tier rappresentano il nucleo di una strategia di trasformazione digitale, in quanto consentono la gestione dell'integrazione dei dati, della messaggistica e dell'Application Programming Interface (API). Le applicazioni middle-tier forniscono inoltre una piattaforma per lo sviluppo e la gestione di applicazioni in grado di coordinare applicazioni enterprise tradizionali e applicazioni distribuite cloud-native.

In modo specifico, una piattaforma applicativa basata su Java™ EE può fornire supporto per le tecnologie correnti e le applicazioni cloud-native nei seguenti modi:

- sfruttando i team esistenti e conoscenze specifiche per le nuove tecnologie;
- preservando le applicazioni legacy e i dati importanti;
- sviluppando nuove applicazioni in parallelo con l'ambiente esistente;
- implementando nuovi processi e architetture.

Ogni azienda seguirà il processo di trasformazione digitale in modo diverso. Ogni organizzazione definisce la propria strategia e i propri obiettivi, ma vi sono principi chiave che permettono di orientarsi verso le scelte giuste e di sfruttare le risorse IT esistenti per ottenere risultati ottimali.



facebook.com/redhatinc
@redhatnews
linkedin.com/company/red-hat

it.redhat.com

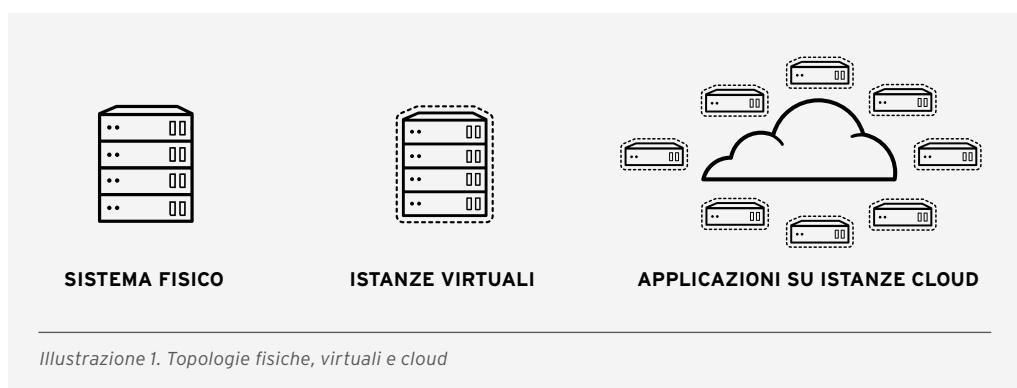
¹ Gens, Frank. *IDC FutureScape: Worldwide IT Industry 2016 Predictions – Leading Digital Transformation to Scale*. IDC, nov. 2015.

² Zetlin, Minda. "How to Balance Maintenance and IT Innovation." *ComputerWorld*, 21 ott. 2013. Web.

³ Fleming, Maureen. *New Middle-Tier Competencies Enabling Digital Transformation*. Rep. IDC, giugno 2016. Web. Sponsorizzato da Red Hat.

UNO SGUARDO AL FUTURO: PIATTAFORMA, PROCESSO, ARCHITETTURA

Nella sua analisi dello stato attuale dello sviluppo software, Al Hilwa, analista IDC, ha sottolineato⁴ che un software efficace è il risultato della combinazione tra architettura software, processi e abilità degli sviluppatori. I reparti IT e i team operativi stanno affrontando cambiamenti significativi in tutti e tre gli ambiti. Con l'introduzione, da parte dell'elaborazione cloud e distribuita, di nuove economie di scala per l'infrastruttura, i tre ambiti fondamentali (architettura, processo e piattaforma) si stanno trasformando per sfruttare gli ambienti cloud.



Piattaforma: cloud

Il cloud computing sarà l'infrastruttura chiave della prossima ondata di innovazioni IT. Il gruppo di analisti IDC stima che, entro il 2020, fino al 70% delle spese totali per le infrastrutture riguarderà i servizi cloud.⁵

La chiave dell'efficacia del cloud sta nella reattività alle variazioni negli ambienti. La virtualizzazione è uno dei principali motori di spinta in ambito IT, poiché astrae l'ambiente operativo dall'ambiente fisico. La virtualizzazione tratta il sistema operativo come un elemento interamente separato dal sistema fisico, consentendo così l'installazione e l'esecuzione di più istanze del sistema operativo mediante lo stesso hardware. Il cloud frammenta ulteriormente l'ambiente, astralendo le applicazioni in esecuzione dal sistema operativo o dall'ambiente fisico sottostante.

I servizi basati sul cloud esistono all'interno di un gruppo distribuito di nodi, senza essere strettamente assegnati a un sistema fisico. Ciò garantisce ridondanza (se un nodo presenta un errore, gli altri nodi portano avanti le operazioni) e scalabilità: è possibile, infatti, creare ed eliminare nuove istanze in base alla richiesta di un servizio specifico.

Questa infrastruttura leggera rende possibile sostenere schemi architettonici altamente decentralizzati (come i microservizi). Gli ambienti cloud generano inoltre efficienza operativa e risparmio in termini economici, massimizzando lo sfruttamento di risorse.

⁴ Hilwa, Al. "The New Developer Landscape – Understanding the Modern Software Developer." Mar. 2016. Presentazione evento IDC.

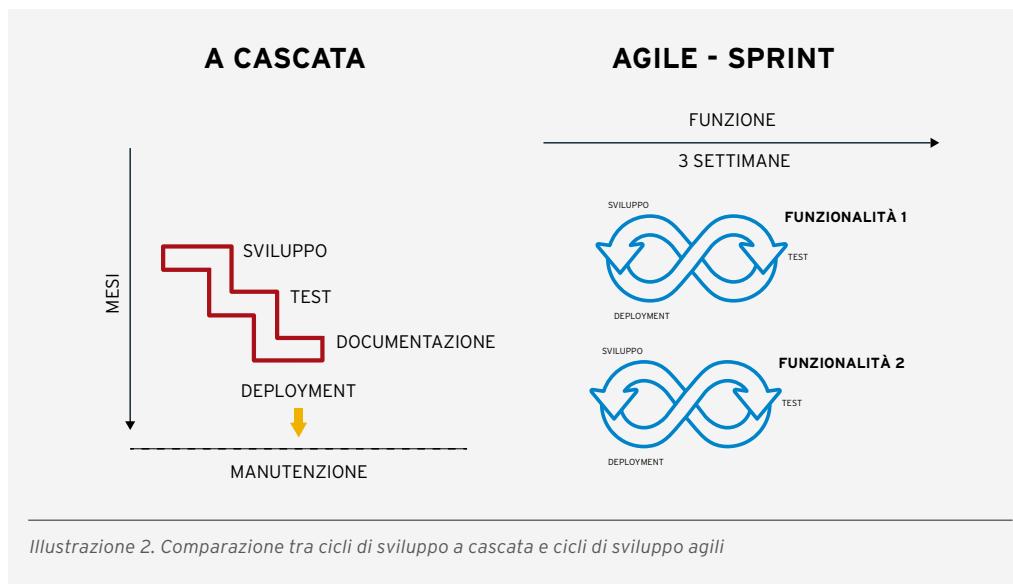
⁵ Ibid., Gens.

Processo: DevOps e agilità

Agilità, Continuous Integration/Continuous Delivery (CI/CD) e DevOps sono elementi chiave della struttura e delle funzioni dei moderni team IT.

L'agilità rappresenta un approccio alla pianificazione e all'esecuzione di un progetto. Un approccio agile, come definito nel relativo Manifesto,⁶ presenta quattro principi chiave:

- persone/team al primo posto;
- collaborazione con gruppi esterni;
- risposta a situazioni mutevoli;
- creazione di un software facile da capire.



I processi agili scompongono le funzioni di progetto in attività gestibili, con l'obiettivo di fornire un'iterazione rapida attraverso le attività e fornendo soluzioni software caratterizzate dall'efficienza di ogni singolo componente. Ciò risulta essere in contrasto con l'approccio confinato e sequenziale dei progetti a cascata. In base a una recente valutazione Gartner, l'approccio agile sta guadagnando popolarità, mentre i progetti a cascata sono scesi sotto il 50%.⁷

Un cambiamento introdotto dagli approcci agili allo sviluppo di progetti consiste nel coinvolgere altri gruppi nell'intero processo di pianificazione e sviluppo, piuttosto che attendere il completamento dello sviluppo. Al termine di uno sprint, il software deve essere funzionale: ciò significa che deve essere testato durante la fase di sviluppo. I team agili includono la fase di test all'interno del ciclo di sviluppo. L'idea di un test continuo si è tradotta nell'integrazione continua, mentre i rilasci rapidi hanno portato al deployment continuo. Oggi vengono impiegati processi continui, anziché fasi distinte e separate.

⁶ Beck, Kent, et al. 2001. "Manifesto for Agile Software Development". Agile Alliance.

⁷ Wilson, Nathan. Modernizing Application Development Primer for 2016. Gartner, 14 gen. 2016. Web.

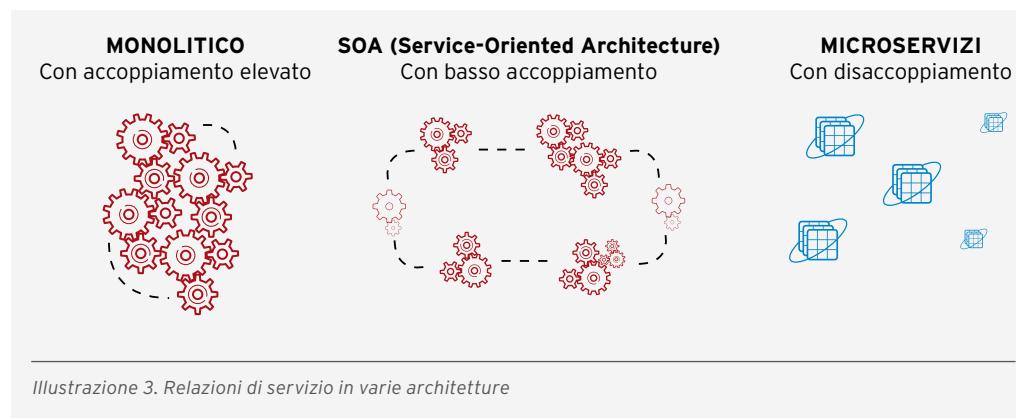
Tuttavia, il ciclo di vita delle applicazioni non finisce alla consegna. Allo sviluppo seguono il deployment e la manutenzione, attività che, in genere, vengono svolte dai reparti IT e dai team operativi, anziché dagli sviluppatori. Se esistesse una netta separazione tra i team, gli sviluppatori potrebbero non comprendere l'ambiente operativo, e i team operativi potrebbero non comprendere la finalità strategica del prodotto. La mancanza di comprensione rischia di generare un ambiente in cui il software o l'infrastruttura producono risultati inferiori alle aspettative. L'unione tra team operativi e di sviluppo è definita DevOps.

ARCHITETTURA: MICROSERVIZI

Il software enterprise ha sempre avuto un'ampia portata. Da un'unica applicazione ci si aspettava la gestione di carichi di picco per un'intera azienda. Qualsiasi funzione richiesta da tale applicazione doveva essere eseguita dall'applicazione stessa. I servizi erano semplici funzionalità di un'applicazione monolitica. Un buon esempio di applicazione monolitica è un database, in cui un singolo database può servire un'intera organizzazione.

Dato che le richieste di elaborazione in ambito enterprise sono diventate più complesse, i requisiti di manutenzione per le applicazioni monolitiche sono cresciuti. Perfino variazioni minime a un'infrastruttura monolitica possono risultare difficili, poiché comportano interventi all'intera infrastruttura. Nasce da qui un nuovo approccio all'architettura definito Service-Oriented Architecture (SOA). Invece di utilizzare un'unica applicazione che compia tutte le operazioni, all'interno di un'architettura orientata al servizio, le funzioni possono essere svolte da diverse applicazioni con basso accoppiamento all'interno di uno schema di integrazione, come avviene in un Enterprise Service Bus (ESB).

L'architettura SOA introduce una complessità aggiuntiva di sistema nell'ambiente generale. Anche se determinati aspetti dell'architettura vengono semplificati (come l'introduzione di nuovi componenti o l'esecuzione di upgrade), l'architettura SOA rischia inoltre di introdurre modifiche a cascata nell'intero ambiente, qualora le interazioni tra i componenti non vengano comprese chiaramente.



Nonostante l'architettura SOA rappresenti un passo nella giusta direzione, è stato sviluppato un modello più maturo, denominato architettura dei microservizi. Le architetture dei microservizi sono simili agli schemi SOA, in quanto i servizi sono specializzati e caratterizzati da un basso accoppiamento, ma i microservizi risultano ulteriormente scomposti. Un'architettura dei microservizi definisce i servizi in modo preciso attraverso:

JAVARESTAILNUMEROUNO

“Java ha davanti a sé un futuro davvero luminoso. I linguaggi e le piattaforme non cambiano dal giorno alla notte. Sono disponibili molte tecnologie interessantissime, in gran parte open source, ma al momento non c'è nulla che possa prendere il posto di Java. Per quanto riguarda il lavoro associato ai microservizi, gran parte si basa su Java. In ambito enterprise, Java sta attraversando una fase di ridefinizione e di reintroduzione nello schema attuale, insieme all'elaborazione cloud ibrida e aperta. Stanno accadendo grandi cose, tutte accomunate dall'open source.”

RICH SHARPLES,
DIRETTORE SENIOR DELLA
GESTIONE PRODOTTI
PER IL MIDDLEWARE,
RED HAT

- uno scopo unico e chiaro;

- parametri ben definiti;

- un'implementazione poliglotta;

I servizi all'interno dell'infrastruttura utilizzano un framework di messaggistica comune, come le API REST (Representational State Transfer), che utilizzano per comunicare l'un l'altro senza difficili transazioni di conversione dati o livelli aggiuntivi di integrazione.

Il framework per la messaggistica consente e favorisce una fornitura più rapida di nuove funzioni e aggiornamenti. Ciascun servizio è indipendente. Un servizio può essere sostituito, ottimizzato o eliminato senza influire su alcun altro servizio nell'infrastruttura. Tale architettura leggera contribuisce a ottimizzare le risorse distribuite o cloud e a supportare la scalabilità dinamica dei singoli servizi.

La legge di Conway e il futuro

Cloud, DevOps e microservizi condividono un tratto comune: la complessità distribuita. Nel 1967, lo sviluppatore Melvin Conway⁸ osservò che il software viene progettato secondo schemi che corrispondono alla struttura di comunicazione dei team coinvolti nel relativo sviluppo. Se la comunicazione è rigida, opaca o incompleta, l'esecuzione del software sarà carente.

Indipendentemente dalla fase della trasformazione digitale in cui si trova un'azienda, il risultato finale rifletterà sempre la cultura dei team IT. Pertanto, prima di passare alla tecnologia vi sono alcune considerazioni⁹ da affrontare in merito alla cultura aziendale:

- stabilire schemi di comunicazione ben compresi dai vari team;
- ridurre al minimo le barriere tra i gruppi;
- promuovere un'infrastruttura e processi burocratici flessibili.

I microservizi, così come qualsiasi architettura complessa distribuita, necessitano di una base solida.

A livello tecnologico, i microservizi non possono esistere senza un'infrastruttura basata sul cloud.¹⁰ A livello organizzativo, le architetture di microservizi si rivelano improduttive in assenza di un ambiente DevOps solido o agile, costruito sulla base della collaborazione, della comunicazione e dell'interfunzionalità tra i team.

IL PASSATO E IL FUTURO DI JAVA EE

Come è possibile indirizzare il proprio attuale reparto IT verso un futuro caratterizzato dalla produttività e dalla complessità delle architetture leggere basate sul cloud?

Fin dalla sua introduzione nel 1995, Java è cresciuto al punto da diventare il linguaggio di programmazione più utilizzato al mondo.¹¹ Oracle ha raggiunto una popolazione globale di sviluppatori Java pari a oltre 9 milioni.¹² Si tratta di circa l'82% degli 11 milioni di sviluppatori professionisti a livello mondiale.¹³

⁸ Conway, Melvin E. "How do Committees Invent?" Datamation, 14 (5): 28-31. aprile 1968.

⁹ DevNation Afternoon General Session. Di Rachel Laycock. California, San Francisco. 27 giugno 2016. Performance. <https://youtu.be/EC2rk9Jh5Ps>

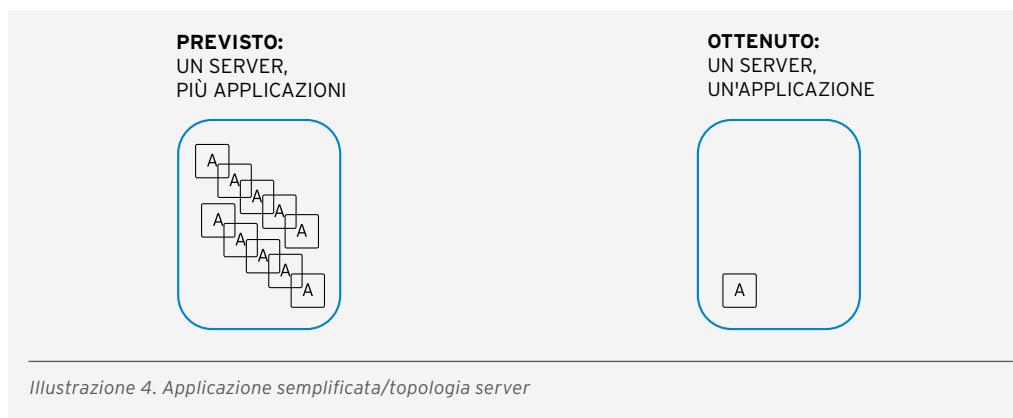
¹⁰ Gens, Frank. IDC FutureScape: Worldwide IT Industry 2016 Predictions – Leading Digital Transformation to Scale. IDC, nov. 2015.

¹¹ "TIOBE Index for August 2016." Ago. 2016. Web.

¹² Beneke, Timothy e Tori Wieldt. JavaOne 2013 Review: Java Takes on the Internet of Things.

¹³ Hilwa, Al. "The New Developer Landscape – Understanding the Modern Software Developer." Mar. 2016. Presentazione evento IDC.

Sun (e successivamente Oracle) hanno sviluppato il linguaggio di programmazione Java ulteriormente, definendo un insieme di standard per le operazioni più diffuse, un set di API e un ambiente runtime per le applicazioni. Questa versione è nota come Java Enterprise Edition (Java EE). I server che implementano le specifiche Java EE sono piattaforme applicative Java e rappresentano il nucleo di svariati ambienti di sviluppo IT. Il modello di programmazione server-client di Java lo ha reso particolarmente adatto ai primi framework presenti in Internet e alle successive applicazioni in ambito enterprise.



Nei primi anni 2000, quando vennero sviluppate le piattaforme applicative, gran parte delle architetture IT fu costruita su applicazioni monolitiche. Le piattaforme applicative Java EE dovevano ospitare svariate applicazioni Java in un'unica posizione centralizzata. Nelle applicazioni pratiche, a causa di fattori tra cui distribuzione del traffico, larghezza di banda, latenza di rete, ridondanza e separazione organizzativa, una piattaforma applicativa spesso ospitava una sola applicazione. Gartner ha indicato¹⁴ queste piattaforme applicative Java come "super piattaforme", sottolineando il fatto che tendevano a sprecare risorse IT.

Uno dei motivi per cui Java è così resiliente come linguaggio e come piattaforma sta nella sua adattabilità. Java e Java EE vengono definiti attraverso una community denominata Java Community Process. La community Java è attiva nello sviluppo di Java EE e in progetti correlati come MicroProfile, Wildfly Swarm e Node.js.

Tale community indirizza Java e le piattaforme Java in una direzione che consente lo sviluppo e il deployment cloud-native di applicazioni leggere e interconnesse. Java EE 6 ha introdotto il concetto dei profili, con un profilo completo adatto ai server web full-scale tradizionali e un profilo web per le applicazioni più leggere. I profili e la modularità sono stati estesi in Java EE 7 e ulteriormente ampliati grazie a diversi progetti realizzati dalle community.

- MicroProfile sta considerando una specifica per ottenere un nuovo profilo Java ridotto rispetto al profilo web. Tale profilo sarebbe più adatto ai microservizi, poiché focalizzato su funzioni come le transazioni e la messaggistica.
- Wildfly Swarm tratta le applicazioni Java come un'immagine di un container, con tutte le librerie e le dipendenze richieste, e solo quelle richieste, inserite in un unico contenitore di ampie dimensioni, come un archivio Java autonomo.

¹⁴ Wilson, Nathan. Modernizing Application Development Primer for 2016. Gartner, 14 gen. 2016. Web.

MODERNIZZAZIONE: SCEGLI IL PERCORSO ADATTO

- Passa al cloud.
- Utilizza una piattaforma applicativa con pari prestazioni a livello di cloud o datacenter.
- Utilizza la tua strategia di business per spingere la strategia IT.
- Decidi quali aspetti dell'infrastruttura modernizzare e come.
- Definisci i tuoi team e la tua cultura aziendale.
- Offri ai tuoi sviluppatori gli strumenti giusti.
- Progetta un ecosistema IT in grado di supportare integrazione, scalabilità e interoperabilità.

Java, come gran parte delle tecnologie, si sta adattando alle infrastrutture cloud con un vantaggio cruciale: il linguaggio chiave è una tecnologia già compresa e utilizzata da milioni di sviluppatori. Le architetture e le implementazioni per le applicazioni sono nuove, tuttavia le capacità necessarie per crearle sono già consolidate.

IL CAMBIAMENTO CHE SFRUTTA AL MASSIMO LE RISORSE

Secondo IDC, il cloud è il cuore dell'ambiente IT della Terza piattaforma.¹⁵ IDC identifica tre epoche nell'evoluzione dell'elaborazione:¹⁶

1. mainframe e personal computer;
2. transazioni basate su Internet e architetture client/server;
3. tecnologie incentrate sulle app (come la mobilità, i social media, l'IoT e i big data) ospitate in un cloud.



Questa Terza piattaforma fa ampio affidamento sulle altre due, ma non è pensata per svolgere le stesse funzioni in modo più efficiente: le utilizza, invece, come punto di partenza per raggiungere nuovi obiettivi.

1. Passare al cloud (con Java EE)

Il cloud computing è al centro di qualsiasi ambiente IT moderno: Ciò è dovuto alla scalabilità del cloud computing, specialmente per quanto riguarda la capacità di aggiungere e contrarre dinamicamente nodi on-demand, con modalità impossibili per i tradizionali sistemi fisici o virtuali.

Con piattaforme tecnologiche come Java EE, è fondamentale che la piattaforma sia in grado di supportare svariati tipi di ambienti, come:

¹⁵ Gens, Frank. *IDC FutureScape: Worldwide IT Industry 2016 Predictions – Leading Digital Transformation to Scale*. IDC, nov. 2015.

¹⁶ IDC. *IDC Predicts the Emergence of “the DX Economy” in a Critical Period of Widespread Digital Transformation and Massive Scale Up of 3rd Platform Technologies in Every Industry*. 4 nov. 2015. Web.

- on-premise;
- cloud pubblico (come Amazon Web Services, Microsoft Azure o Google);
- cloud privato (come un cloud privato OpenStack®);
- container;
- servizi in hosting.

Le applicazioni devono essere in grado di funzionare allo stesso modo in ogni ambiente. Alcune piattaforme applicative potrebbero non funzionare in tutti gli ambienti richiesti, oppure potrebbero non offrire le stesse funzioni nei diversi ambienti. La scelta della giusta piattaforma Java EE assicura l'interoperabilità cruciale per gli ambienti misti.

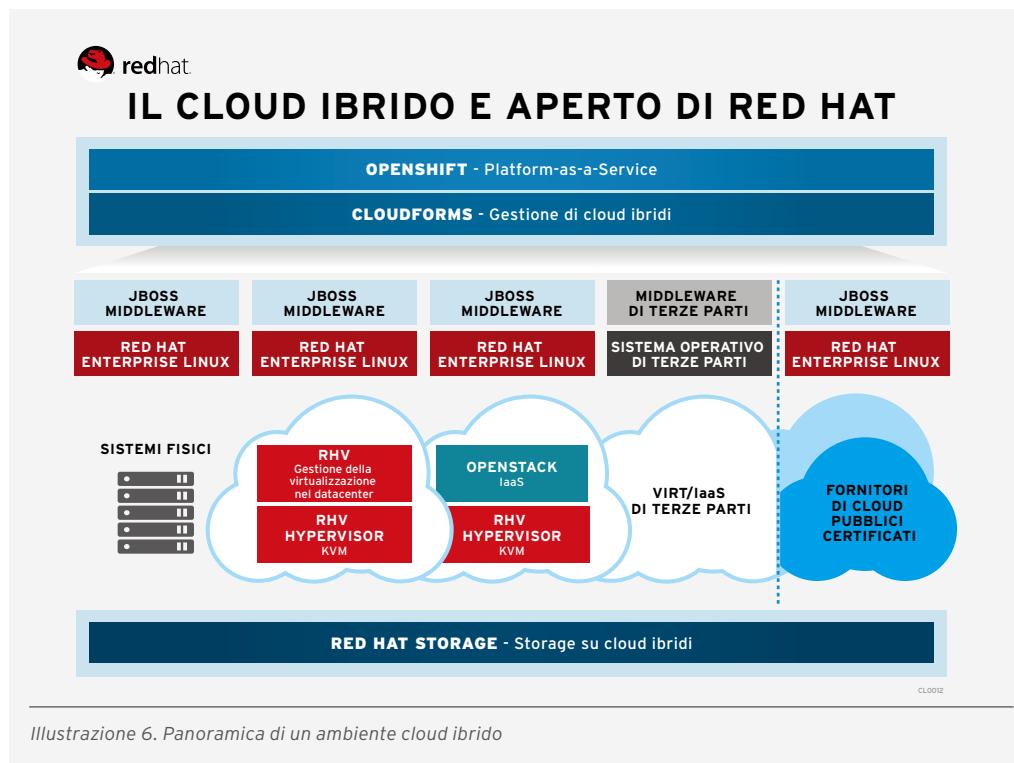


Illustrazione 6. Panoramica di un ambiente cloud ibrido

Oltre alla complessità del funzionamento in ambienti ibridi, vi è un costo economico legato al funzionamento in ambienti multipli. Gartner ha identificato, nella gestione delle licenze software, una potenziale riduzione dei costi per i reparti IT,¹⁷ ritenendola l'area più sottovalutata per la riduzione dei costi. La gestione dei costi dei software può risultare complessa, se si considerano le diverse licenze in ambito on-premise, virtuale o cloud e i servizi di abbonamento aggiuntivi per vari tipi di supporto.

¹⁷ McGittigan, Jim e Sanil Solanki. The Gartner Top 10 Recommended IT Cost Optimization Ideas, 2016. Tech. no. G00301094. Gartner, 29 feb. 2016. Web.

SUGGERIMENTO: DECIDI TU FIN DOVE ARRIVARE.

Non tutte le infrastrutture IT devono passare a un'architettura di microservizi in stile startup per raggiungere i propri obiettivi di trasformazione digitale. Nella migrazione al cloud, utilizzare deployment basati sui container o presentare nuove iniziative IoT o mobili può essere il passo giusto per realizzare i propri piani strategici.

2. Porre domande esistenziali

Il modo per determinare l'opzione migliore per il proprio ambiente è bilanciare il rischio di modifica con gli obiettivi strategici. I requisiti sono diversi per ogni organizzazione. La strategia digitale dovrebbe riflettere la strategia business.

- Quali sono gli obiettivi dell'organizzazione o la direzione strategica?
- Si dispone delle abilità e delle risorse necessarie per seguire la direzione strategica scelta?
- Si dispone di un ambiente in grado di sostenere le applicazioni aziendali?

IL TUO PERCORSO VERSO I MICROSERVIZI

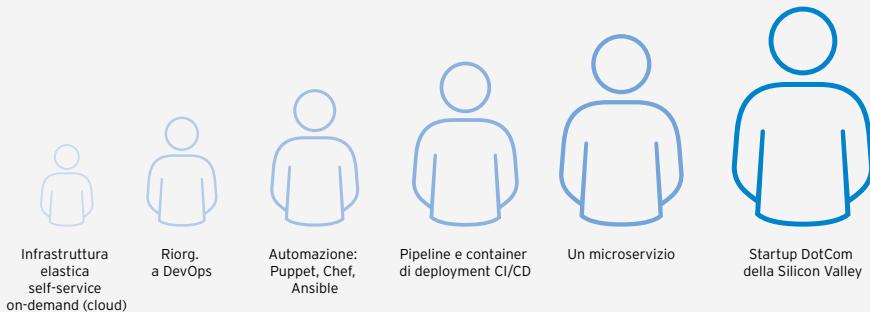


Illustrazione 7. Stadi dell'evoluzione dell'ambiente

3. Decidere cosa rinnovare e come

La domanda centrale della trasformazione digitale, secondo Peter Marston di IDC, è come affrontare la modernizzazione delle applicazioni. In base alle ricerche di IDC, circa il 40% delle organizzazioni ha stabilito come priorità numero uno per l'IT la modernizzazione delle applicazioni o intende realizzarla entro i prossimi tre anni.¹⁸

Vi sono svariati approcci alla modernizzazione:

- ricorrere al retrofit o a nuove architetture che rendano le applicazioni esistenti adatte ad un ambiente moderno;
- effettuare la migrazione di un'applicazione da un ambiente a un altro;
- sostituire le applicazioni esistenti con nuove applicazioni;
- creare ambienti paralleli.

Ogni azienda sceglierà l'approccio a seconda della direzione del percorso di adozione dei microservizi.

Un obiettivo è ridurre il focus dell'IT sulla manutenzione. È necessario suddividere la spesa IT e la concentrazione delle risorse in modo equilibrato, tra progetti nuovi ed esistenti. Tuttavia, quasi tre quarti (72%) della spesa IT sono destinati a progetti operativi e di manutenzione. La maggioranza dei CTO (63%) pensa che sia troppo elevata.¹⁹

¹⁸ Marston, Peter. *Ten Criteria to Use for Application Modernization Service Provider Selection*. Rep. no. IDC #US41012716. IDC, feb. 2016. Web.

¹⁹ Zetlin, Minda. "How to Balance Maintenance and IT Innovation." ComputerWorld, 21 ott. 2013. Web.

Un ambiente Java EE basato sul cloud risulta utile ai fini di alleviare i costi di manutenzione. Applicazioni nuove e esistenti possono essere eseguite sulla stessa piattaforma, poiché in grado di gestire sia soluzioni monolitiche sia microservizi. Cosa più importante, è inoltre possibile effettuare la migrazione di applicazioni Java tra ambienti diversi.

Tale portabilità riduce i costi di transazione e i rischi di migrazione, pur fornendo un percorso rivolto alla modernizzazione delle applicazioni. Ogni organizzazione può così modificare la propria infrastruttura in modo incrementale, per prima cosa ottenendo i vantaggi economici del cloud e, quindi, le efficienze operative dei container senza doversi vincolare a microservizi o ad architetture altamente distribuite, a meno che non risultino adatte alla propria strategia.

4. Investire sui propri team

Una delle sfide principali che i CIO si trovano ad affrontare è la carenza percepita di talenti IT in grado di fare la differenza nella nuova Terza piattaforma. Un'analisi dell'Harvard Business Review sulla crisi dei talenti IT rivela che il novantacinque per cento dei CIO ritiene che la carenza di competenze impedisca loro di rispondere alle sfide strategiche o IT.²⁰ Le aree fonti di preoccupazione si concentrano intorno alla Terza piattaforma IDC:

- big data e analisi (36%);
- architetti, in ambito enterprise (27%) o tecnico (24%);
- sviluppo (27%);
- sviluppo mobile (24%);
- strategia IT (22%).

Nigel Fenwick di Forrester Research ha rilevato che, spesso, le aziende del settore IT cercano di raggiungere ogni obiettivo IT ricorrendo a una soluzione personalizzata, nonostante esistano approcci strategici migliori. "Abbiamo investito milioni di dollari nella personalizzazione del software al fine di supportare capacità generiche. Ciò ha reso l'IT più complesso, ha reso le interfacce più difficili da usare, ha ridotto l'agilità IT e incrementato i costi."²¹

Invece di investire tempo e risorse tutte le direzioni, è consigliabile semplificare e focalizzare la propria strategia IT seguendo alcune indicazioni:

- identificare due o tre competenze chiave al fine di stabilire un obiettivo strategico;
- utilizzare soluzioni basate su standard, anziché soluzioni personalizzate per tutti gli altri requisiti;
- selezionare soluzioni di facile manutenzione.

La maggiore carenza di competenze identificata dai CIO riguarda i processi, non la tecnologia. L'utilizzo di standard aperti e di tecnologie comuni come Java può contribuire a ridurre il tempo che occorre per acquisire competenze. L'implementazione di una piattaforma applicativa Java EE come piattaforma di sviluppo consente ai reparti IT di sfruttare le proprie conoscenze ed esperienze per sviluppare progetti in nuove aree. L'utilizzo diffuso di Java nelle community di sviluppatori a livello professionale amplia il bacino di potenziali collaboratori.

5. Fornire agli sviluppatori gli strumenti giusti

La presenza, nella piattaforma tecnologica, di strumenti integrati e realizzati per le fasi di sviluppo, aiuta a rendere il ciclo complessivo di sviluppo più uniforme, specialmente con il test o l'automazione integrati. Vi sono svariate aree chiave in cui gli sviluppatori possono vedere vantaggi diretti:

²⁰ I.T. Talent Crisis: Proven Advice from CIOs and HR Leaders. Tech. Harvard Business Review Analytic Services, luglio 2016. Web. Sponsorizzato da Red Hat

²¹ Zetlin, Minda. "How to Balance Maintenance and IT Innovation." ComputerWorld, 21 ott. 2013. Web.

- moduli di testing integrato per CI/CD;
- automazione per deployment;
- kit di strumenti per sviluppatori;
- caricamento delle classi personalizzato;
- prestazioni.

Anche funzionalità minori legate alle prestazioni possono avere un impatto significativo. Ad esempio, tempi più rapidi di avvio al momento di eseguire il deployment delle applicazioni possono consentire di risparmiare svariate ore alla settimana e tradursi in una maggiore produttività degli sviluppatori, poiché è possibile riavviare un'applicazione più volte al giorno, durante il normale processo sviluppo.

Altre funzioni, come il test integrato, sono cruciali per DevOps o CI/CD. Un'azienda IT australiana ha incrementato la produttività degli sviluppatori del 15% semplicemente caricando i moduli di test all'interno del processo di deployment.²² L'integrazione tra sviluppo, test e operazioni ha contribuito a migliorare la qualità del codice e a ridurre il ciclo di vita dello sviluppo.

6. Costruire un ecosistema

Secondo Maureen Fleming, analista IDC, la forza della trasformazione digitale è l'integrazione.²³ La Terza piattaforma presenta molti aspetti che possono collaborare tra loro, tra cui:

- virtualizzazione, cloud pubblici e cloud privati;
- container e orchestrazione;
- virtualizzazione dati;
- cache di memoria e storage;
- protocolli di messaggistica multipli;
- formati di dati diversi provenienti da fonti diverse;
- strumenti di gestione e deployment;
- automazione dei test;
- automazione dei processi aziendali.

La progettazione dell'effettiva piattaforma rappresenta soltanto una delle fasi all'interno di un'iniziativa di trasformazione digitale. L'ecosistema che racchiude la piattaforma e le relative applicazioni è molto più vasto.

Peter Martson di IDC consiglia di considerare l'ecosistema allargato di un provider di piattaforme applicative come parte di una strategia più ampia rivolta alla modernizzazione delle applicazioni.²⁴ Un fornitore di soluzioni in grado di offrire esperienza e linee guida attraverso le proprie tecnologie e competenze può risultare utile sotto molti aspetti, dalla pianificazione dell'architettura al cloud provisioning.

²² Fleming, Maureen e Matthew Marden. *The Business Value of Red Hat JBoss Enterprise Application Platform*. Tech. no. #257256. IDC, luglio 2015. Web. Sponsorizzato da Red Hat.

²³ Fleming, Maureen. *Integration Is a Core Competency of Digital Transformation*. Tech. no. IDC #US41293916. IDC, maggio 2016. Web. Sponsorizzato da Red Hat.

²⁴ Marston, Peter. *Ten Criteria to Use for Application Modernization Service Provider Selection*. Rep. no. IDC #US41012716. IDC, feb. 2016. Web.

WHITE PAPER Una piattaforma per la modernizzazione delle applicazioni Java

CONCLUSIONE

La trasformazione digitale è un approccio strategico all'IT che considera l'infrastruttura IT e i dati come un prodotto potenziale per i clienti. Per ottenere la trasformazione digitale è necessario distogliere l'attenzione dalle applicazioni con focus interno, che svolgono funzioni business interne, e trattare le applicazioni come potenziali prodotti, cercando nuovi modi di utilizzare i dati e le fonti di dati, e considerando nuovi modi di attrarre clienti.

Il passaggio alla trasformazione digitale richiede tre interventi:

- stabilire chiaramente obiettivi organizzativi e strategia;
- implementare processi solidi, collaborativi e comunicativi tra team funzionali;
- creare un'architettura distribuita disciplinata e ben compresa.

Le applicazioni Java rappresentano da due decenni una tecnologia centrale per le aziende. Tale ricchezza di dati, funzionalità e conoscenza è fondamentale per le aziende. Oggi, con le applicazioni Java, lo sviluppo può avvenire su una piattaforma che supporti simultaneamente architetture cloud-native e applicazioni esistenti. In ambito aziendale, la combinazione tra monoliti tradizionali e deployment di applicazioni basate sul cloud consente di sfruttare la conoscenza e le risorse esistenti, e di passare, allo stesso tempo, a modelli di applicazioni più recenti.

L'utilizzo di una piattaforma applicativa Java rivolta al cloud offre diversi vantaggi alle organizzazioni:

- riduce l'impatto delle carenze in ambito di competenze e tecnologico all'interno del reparto IT;
- amplia i potenziali pool di risorse;
- fornisce una strategia di migrazione a ponte per i carichi di lavoro esistenti, pur consentendo lo sviluppo in ambienti nuovi;
- fornisce un ambiente per l'introduzione di altre tecnologie per l'integrazione, la gestione dati e altre iniziative strategiche.

Una piattaforma applicativa Java appropriata può non solo gestire le applicazioni enterprise esistenti, ma diventare un metodo per raggiungere la trasformazione digitale pianificata dal team esecutivo, massimizzando allo stesso tempo le risorse IT esistenti e portando avanti progetti critici di manutenzione.

INFORMAZIONI SU RED HAT

Red Hat è il leader mondiale nella fornitura di soluzioni software open source e si avvale di un approccio community-based per offrire tecnologie cloud, Linux, middleware, storage e di virtualizzazione caratterizzate da affidabilità e prestazioni elevate. L'azienda offre inoltre servizi di supporto, formazione e consulenza per i quali ha ottenuto diversi riconoscimenti. Principale punto di riferimento in una rete globale di aziende, partner e community open source, Red Hat consente di creare tecnologie specifiche e innovative che garantiscono libero accesso alle risorse per la crescita e preparano i clienti al futuro dell'IT.



facebook.com/redhatinc
@redhatnews
linkedin.com/company/red-hat

it.redhat.com
INC0441724_1016

**EUROPA, MEDIO ORIENTE
E AFRICA (EMEA)**
00800 7334 2835
it.redhat.com
europe@redhat.com

TURCHIA
00800-448820640

ISRAELE
1-809 449548

EAU
8000-4449549

Copyright © 2016 Red Hat, Inc. Red Hat, Red Hat Enterprise Linux, il logo Shadowman e JBoss sono marchi di Red Hat, Inc. registrati negli Stati Uniti e in altri Paesi. Linux® è un marchio registrato di Linus Torvalds negli Stati Uniti e in altri Paesi.

Il marchio denominativo OpenStack® e il logo OpenStack sono marchi commerciali/di servizio di proprietà della OpenStack Foundation, registrati negli Stati Uniti e in altri Paesi, e sono da utilizzarsi previa autorizzazione della OpenStack Foundation. Red Hat non ha rapporti di affiliazione con la OpenStack Foundation o con la community di OpenStack, né riceve da esse sponsorizzazioni o finanziamenti.