

# Integrazione agile

Un approccio moderno  
all'integrazione ibrida aziendale

# Indice

- 1 Panoramica
- 2 L'integrazione è cambiata
- 3 Il caso per l'integrazione agile
- 4 Piattaforme di integrazione ibrida
- 5 IBM Cloud Pak for Integration

## Panoramica

Oggi, come non mai, le organizzazioni richiedono l'integrazione di applicazioni, dati e sistemi per trasformare i modelli di interazione e ottimizzare le prestazioni del canale, pur mantenendo ancora la business continuity. Gli approcci tradizionali all'integrazione hanno costi elevati e non riescono a stare al passo con la continua crescita della domanda.

La tua azienda ha bisogno di un approccio all'integrazione agile e moderno. Dovrebbe dotare i team estesi di strumenti per creare integrazioni, utilizzare efficacemente una serie completa di stili e funzionalità di integrazione e incrementare la produttività nel suo complesso. Le organizzazioni che perseguono la trasformazione digitale devono adottare modi nuovi per utilizzare e implementare le tecnologie di integrazione, per poter ottenere rapidamente i vantaggi garantiti dalle moderne piattaforme cloud.

Molte organizzazioni hanno iniziato ad adottare le tecniche per lo sviluppo agile di applicazioni, ad esempio l'architettura di microservizi, e ora cominciano a riscontrare i vantaggi di tale scelta.

Questo smart paper deriva da un **IBM Redbook** che esamina i meriti di quella che definiamo integrazione agile – un approccio allineato ai microservizi, decentralizzato e basato su contenitori per l'integrazione di soluzioni, che garantisce l'agilità, la scalabilità e la resilienza richieste dalla trasformazione digitale.

L'integrazione agile consente di liberarsi delle pesanti architetture di integrazione centralizzate, che non riescono a supportare la domanda. Aumenta il ritmo con il quale si possono creare integrazioni, tramite la semplificazione e l'automazione. L'integrazione, quindi, diventa un abilitatore cruciale di innovazione, piuttosto che un fattore di ostacolo.

Questo nuovo approccio si estende dalle persone e i processi all'architettura e alla tecnologia, utilizzando efficacemente procedure e funzionalità moderne, che includono API, architettura di microservizi, modalità cloud-native di progettazione, DevOps, architettura orientata dagli eventi e infrastruttura basata su contenitori.

Per approfondire, leggi il nostro IBM Redbook, *Accelerating Modernization with Agile Integration*

Scarica



## L'integrazione è cambiata

**Le aziende stanno guardando alla trasformazione digitale come chiave per la creazione di nuove esperienze personalizzate e connesse attraverso una rete di applicazioni, che utilizza efficacemente tutti i tipi di dati.**

Unificare processi e fonti di informazioni al momento giusto e nel giusto contesto è, a dir poco, difficile – soprattutto se si pensa alla massiccia adozione di applicazioni di business SaaS (software-as-a-service). Dati provenienti da nuove origini dati devono essere inseriti nei processi di business, per ottenere un vantaggio competitivo.

## Il valore dell'integrazione agile per la trasformazione digitale.

Con una tale quantità di dati e una distribuzione così ampia, la velocità con cui si riescono a raggruppare tali dati e si rendono disponibili tramite API ed eventi è fondamentale per il successo delle innovazioni moderne. L'integrazione agile accelera la capacità di:

- ◇ **Fare fronte alla disparità:** Accedere ai dati da qualsiasi sistema, in qualsiasi formato, e unificarli per fornire maggiore insight, indipendentemente da quanto diversificato sia il proprio panorama multicloud.
- ◇ **Usufruire delle competenze in materia di sistemi di back-end:** L'integrazione moderna include intelligence integrata relativa a protocolli e formati di dati complessi e incorpora intelligence sugli oggetti effettivi, sulle funzioni di business e dei sistemi di back-end.
- ◇ **Abilitare l'innovazione con i dati:** I progettisti di soluzioni di applicazioni attuano gran parte della loro innovazione combinando dati oltre i loro confini e creando da essi nuovo significato, una caratteristica particolarmente evidente nell'architettura dei microservizi.
- ◇ **Ricavare artefatti di livello aziendale:** I flussi di integrazione ricavano un'enorme quantità di valore dalla ricchezza del runtime di integrazione sottostante, che include funzioni di livello aziendale per il ripristino in caso di errore, la tolleranza di errore, l'acquisizione di log, l'analisi delle prestazioni e molto altro ancora.

Il panorama dell'integrazione sta cambiando per stare al passo con le richieste del mercato e delle aziende in materia di elaborazione, ma come si può passare dalla SOA (service-oriented architecture) e dagli ESB (enterprise service bus) ad un approccio di integrazione agile, basato su contenitori e moderno?

Per promuovere nuove esperienze cliente, le organizzazioni devono attingere a una serie in continua crescita di applicazioni, processi e fonti di informazioni – che, nel complesso, aumentano in modo significativo l'esigenza aziendale di funzionalità di integrazione e gli investimenti in tal senso.

## Il percorso fino a questo punto – SOA e il modello ESB

Per poter guardare avanti, al futuro dell'integrazione agile, dobbiamo capire cosa è successo prima. I modelli SOA sono emersi all'inizio del millennio e l'ampia accettazione iniziale degli standard SOA ha annunciato un brillante futuro, in cui ogni sistema avrebbe potuto rilevare qualsiasi altro sistema e comunicare con esso, utilizzando modelli di esposizione sincrona.

La SOA veniva implementata normalmente utilizzando il modello ESB, che offre connettività sincrona standardizzata a sistemi di back-end, in genere tramite servizi Web. Anche se molte aziende hanno implementato con successo il modello ESB, la sua popolarità ha creato numerose difficoltà.

- ❖ I modelli ESB spesso formavano un'infrastruttura singola per tutta l'azienda, con decine o centinaia di integrazioni installate su un cluster di server di produzione. Sebbene il modello ESB non richieda una forte centralizzazione, le topologie implementate quasi sempre seguivano questo schema.
- ❖ I modelli ESB centralizzati spesso non riuscivano a produrre i risparmi significativi che le aziende speravano. Un numero esiguo di interfacce poteva essere riutilizzato da un progetto a un altro, eppure la creazione e la manutenzione delle interfacce aveva costi proibitivi per l'ambito di un qualsiasi progetto da intraprendere.
- ❖ L'architettura SOA era più complessa della mera implementazione di un ESB, in particolare dal punto di vista di chi avrebbe finanziato un programma esteso a tutta l'azienda. Iniziative inter-aziendali quali la SOA e l'ESB sottostante erano difficili da finanziare e spesso tale finanziamento si applicava solo a servizi che sarebbero stati riutilizzabili fino a coprire il costo della creazione.

Il risultato è stato che la creazione di servizi da parte di un team di specialisti SOA si è trasformata talvolta in un collo di bottiglia per i progetti, invece di essere un abilitatore.

Nonostante queste difficoltà, il modello ESB centralizzato apporta comunque alcuni vantaggi, in particolare per le organizzazioni con un team di integrazione estremamente qualificato e un basso tasso di attrito e che ricevono un numero di nuovi requisiti di integrazione prevedibile e gestibile.

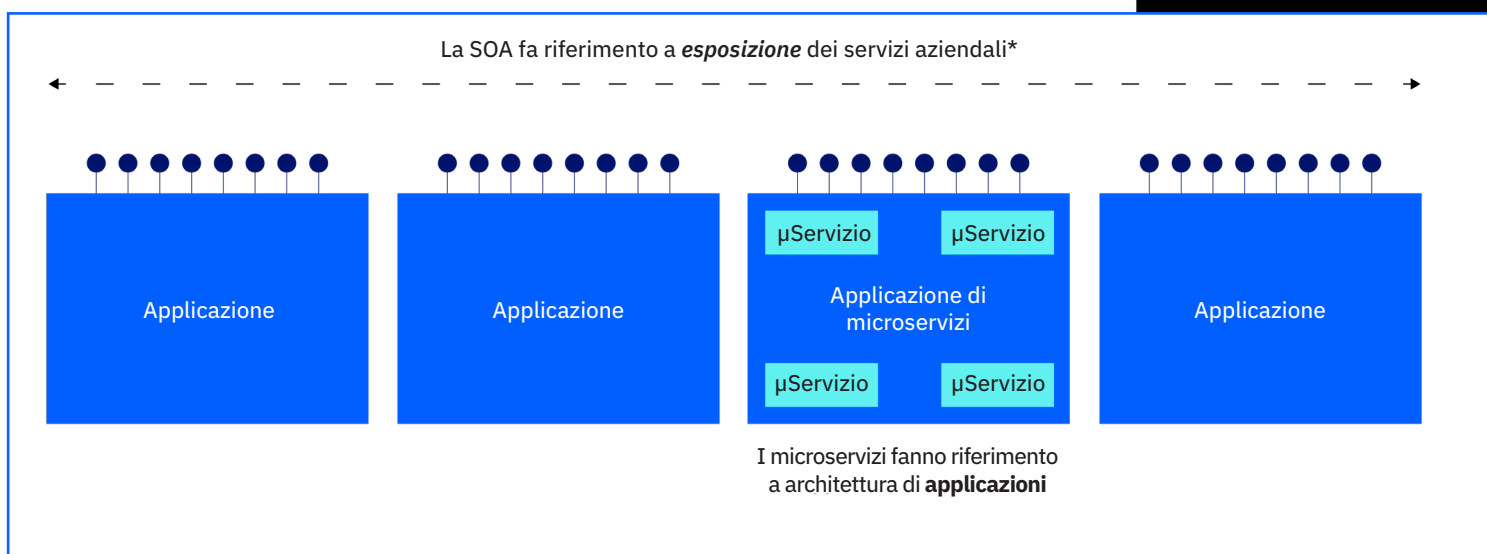
Un singolo ESB centralizzato certamente semplifica la congruenza e la governance dell'implementazione. Tuttavia, molte organizzazioni devono gestire requisiti fluidi e dinamici e sono sotto pressione per la necessità di implementare l'integrazione utilizzando tecnologie cloud-native e metodi agili, simili a quelli che vengono utilizzati in altri reparti dell'organizzazione. Un tipico esempio consiste nel passaggio ad un'architettura di microservizi, tipicamente rilevato nello spazio di sviluppo delle applicazioni. Per questo motivo, il passaggio fondamentale a un'implementazione più granulare di integrazioni è essenziale per l'agilità.



## SOA rispetto ad un'architettura di microservizi

Anche se la SOA e un'architettura di microservizi sembrano simili e vengono spesso confuse, in effetti, si tratta di concetti completamente separati.

L'utilizzo della SOA e del modello ESB associato rappresenta un'iniziativa estesa a tutta l'azienda per rendere i dati e le funzioni nei systems of record immediatamente disponibili per le nuove applicazioni. I progettisti creano interfacce sincrone, riutilizzabili, come ad esempio servizi Web e API RESTful, per esporre i systems of record nell'azienda, quindi, applicazioni nuove e innovative possono essere create più rapidamente, incorporando dati da più sistemi in tempo reale.



Un'architettura di microservizi, d'altro canto, è un modo di scrivere una singola applicazione, come fosse una serie di componenti di dimensioni più ridotte (microservizio), per rendere l'applicazione più agile, scalabile e resiliente.

Riepilogando, la SOA riguarda l'integrazione in tempo reale tra applicazioni, mentre un'architettura di microservizi riguarda il modo in cui le applicazioni sono create internamente.

## Il caso per l'integrazione agile

Perché i concetti legati ai microservizi sono diventati così famosi nello spazio delle applicazioni? Perché forniscono un approccio alternativo alla strutturazione delle applicazioni, che abilita una maggiore agilità.

Piuttosto che intendere un'applicazione come un grande silo di codice in esecuzione sullo stesso server, i microservizi possono aiutare a creare applicazioni composte da una raccolta di componenti più piccoli, che si eseguono in modo completamente indipendente.

Un'architettura di microservizi consente di ottenere tre importanti vantaggi:

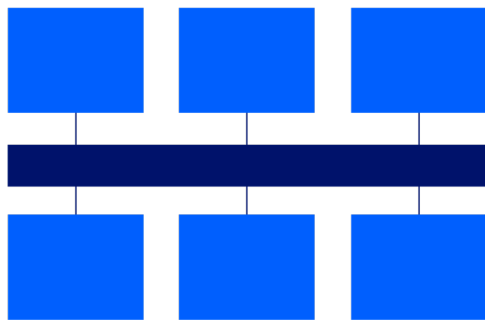
- ❖ **Maggiore agilità:** I microservizi hanno dimensioni sufficientemente ridotte da essere completamente conosciuti dal team proprietario e modificati in modo indipendente.
- ❖ **Scalabilità elastica:** L'utilizzo delle risorse può essere completamente collegato al modello di business.
- ❖ **Resilienza separata:** Con l'adeguata separazione, le modifiche a un microservizio non influiranno sugli altri in fase di runtime.

Tenendo questi vantaggi a mente, cosa succederebbe se ripensassimo l'integrazione, che generalmente viene implementata in silo centralizzati, con una nuova prospettiva basata sull'architettura di microservizi? Questo è ciò che definiamo una "integrazione agile."

L'integrazione agile viene definita come "un'architettura basata su contenitori, decentralizzata e allineata ai microservizi per le soluzioni di integrazione."

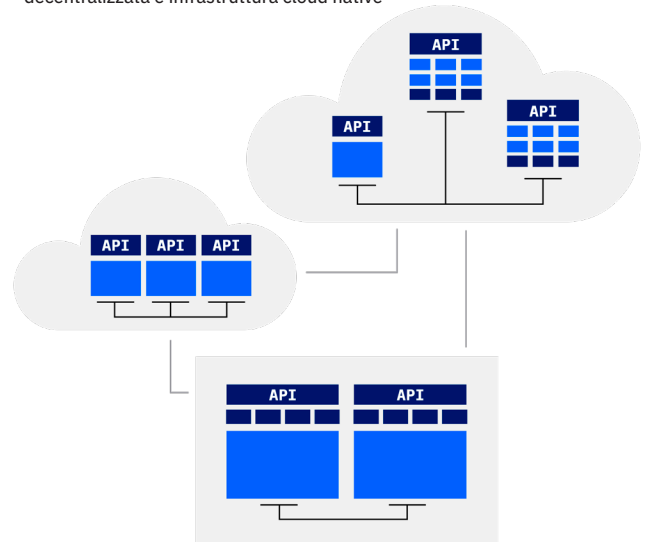
### Architettura di integrazione tradizionale

Provisioning e amministrazione centralizzati



### Architettura di integrazione agile

Implementazione granulare, proprietà decentralizzata e infrastruttura cloud native



Ci sono tre aspetti correlati, ma separati, che riguardano l'integrazione agile:



### **Persone e processo:**

Proprietà dell'integrazione decentralizzata. Come si dovrebbe adeguare la struttura dell'organizzazione per utilizzare più efficacemente un approccio più autonomo e garantire ai team delle applicazioni maggior controllo sulla creazione e l'esposizione della loro propria integrazione e sul modo in cui sono esposti come API, messaggi ed eventi?



### **Architettura:**

Implementazione dell'integrazione granulare. Cosa si potrebbe ottenere se si decidesse di suddividere le integrazioni presenti nell'ESB compartimentalizzato in implementazioni separate, con manutenzione e scalabilità indipendenti. Qual è il modo più semplice in cui queste integrazioni separate potrebbero essere messe a disposizione, in modo coerente e sicuro, all'interno e all'esterno dell'azienda, utilizzando API ed eventi?



### **Tecnologia:**

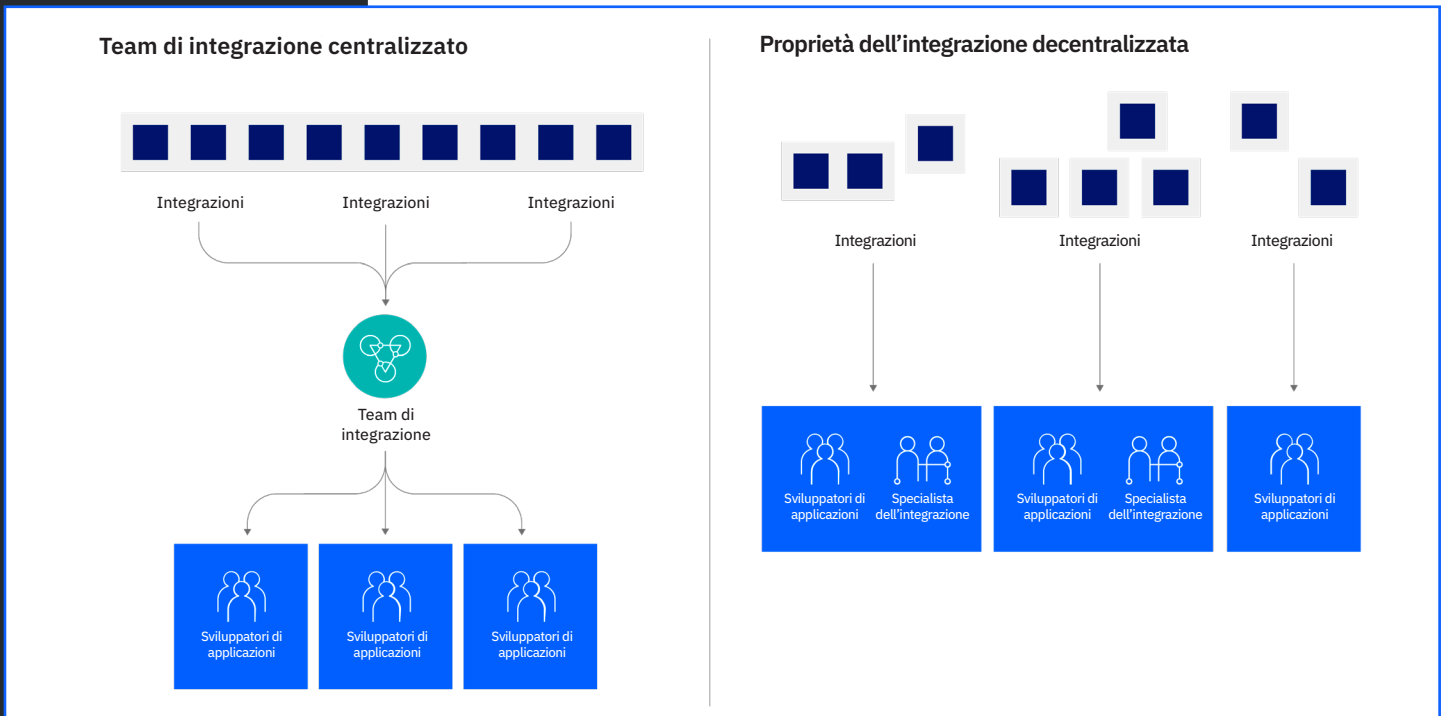
Infrastruttura di integrazione cloud-native. Come si può utilizzare nel modo più efficace un' infrastruttura cloud-native, ad esempio i contenitori, per migliorare la produttività, la congruenza operativa e la portabilità, sia per le applicazioni che per le integrazioni, in ambienti ibridi e multicloud?



## **Persone e processo: proprietà dell'integrazione decentralizzata**

La SOA tendeva a portare alla creazione di un'infrastruttura e di team di integrazione centralizzati per implementare il livello di servizio. Il team centrale conosceva bene la tecnologia di integrazione, ma spesso non comprendeva le applicazioni che stava integrando, quindi, la conversione dei requisiti poteva risultare lenta o soggetta ad errori. Questa dipendenza ha creato un continuo attrito data la velocità di esecuzione possibile dei processi.

Molte organizzazioni avrebbero preferito che i team delle applicazioni possedessero la propria integrazione, ma la tecnologia e l'infrastruttura in quel momento non lo consentivano. Il passaggio ad un'integrazione agile, con implementazione cloud-native, più granulare, rende possibile semplificare e automatizzare gran parte della pipeline di sviluppo, test e implementazione. Questo, a sua volta, consente di distribuire la responsabilità della creazione e manutenzione delle integrazioni al di là del team centrale.



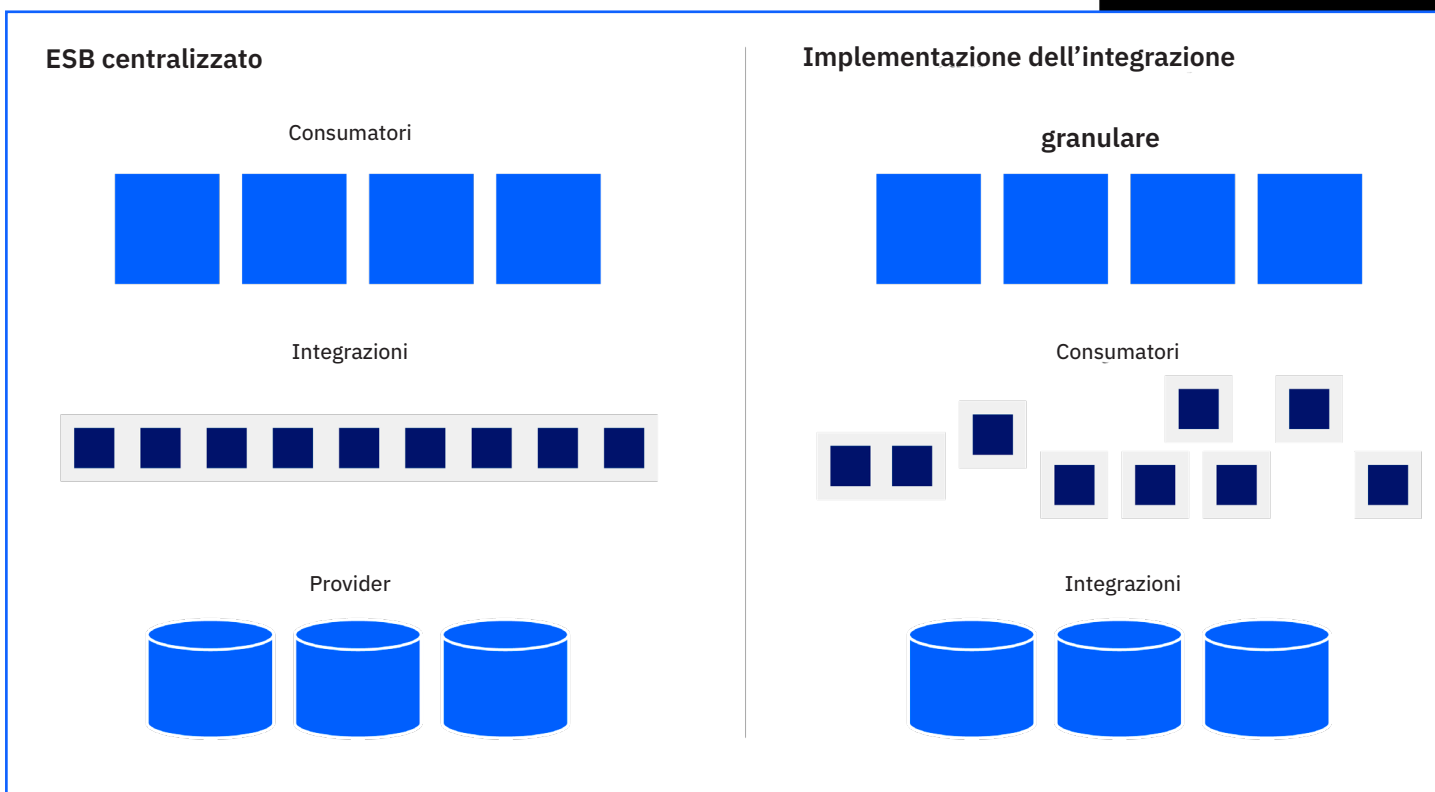
Questo approccio procede di pari passo con la maggiore importanza della gestione delle API. I team devono essere in grado di gestire l'esposizione delle proprie API, piuttosto che fare riferimento ad un team centralizzato di specialisti di gateway API per farlo. La gestione delle API consente ai team delle applicazioni di eseguire autonomamente tale configurazione e anche di popolare i portali dello sviluppatore con definizioni delle API e degli eventi, in modo che gli utenti di questi elementi possano individuarli rapidamente ed eseguire automaticamente l'onboarding per utilizzarli.

Inoltre, applicazioni basate su microservizi spesso preferiscono comunicare internamente, utilizzando eventi. La maggiore separazione aumenta ulteriormente l'autonomia del team di apportare modifiche in modo indipendente. L'integrazione agile prende anche in considerazione il modo migliore di consentire ai team di pubblicare e utilizzare flussi di eventi all'interno e attraverso i confini delle applicazioni.

## Architettura: implementazione dell'integrazione granulare

L'integrazione tradizionale è caratterizzata dall'implementazione centralizzata di integrazioni nel modello ESB. In questo modello, tutte le integrazioni vengono implementate in un'unica coppia molto consolidata di server di integrazione e possono dare luogo a colli di bottiglia per i progetti. Qualsiasi implementazione su server condivisi corre il rischio di destabilizzare le interfacce critiche esistenti. Inoltre, team che eseguono singoli progetti non possono decidere in modo indipendente di aggiornare il middleware dell'integrazione per ottenere accesso a nuove funzioni.

Utilizzando gli stessi concetti su cui si basa un'architettura di microservizi, potremmo suddividere l'ESB esteso a tutta l'azienda in componenti dedicati, più piccoli e più gestibili. Tali modelli di "implementazione dell'integrazione granulare" offrono contenitori specializzati e di dimensioni adeguate, che migliorano agilità, scalabilità e resilienza e che differiscono molto rispetto ai modelli ESB centralizzati del passato. Usufruendo dell'infrastruttura cloud native, di cui discuteremo più avanti, sarà molto più facile configurare pipeline automatizzate, la cui esecuzione non richiede competenze specialistiche. Questo semplifica notevolmente il processo meccanico di implementazione, lasciando a disposizione un tempo maggiore per focalizzarsi sul valore di business dell'integrazione.






*Confronto semplificato tra un ESB centralizzato e un'implementazione dell'integrazione granulare.*

L'implementazione dell'integrazione granulare attinge ai vantaggi dell'architettura di microservizi. Esaminiamo di nuovo quelli che abbiamo definito come vantaggi dei microservizi, alla luce dell'implementazione dell'integrazione granulare.

Scopri la fase in cui ti trovi nel percorso di modernizzazione dell'integrazione e identifica i passi successivi da intraprendere per realizzare un'integrazione agile.

Esegui la valutazione

- 
**Agilità:** Team differenti possono lavorare sulle integrazioni in modo indipendente, senza dover delegare tutto a un'infrastruttura o gruppo centralizzati, il che potrebbe rapidamente trasformarsi in un collo di bottiglia. Modifiche a singoli flussi di integrazione possono essere ricompilate e implementate in modo indipendente da altri flussi, per consentire un'applicazione più sicura delle modifiche e accelerare al massimo il tempo necessario per la produzione.
- 
**Scalabilità:** Anche i problemi operativi sono automatizzati, in modo che i singoli flussi possano essere ridimensionati in modo indipendente, usufruendo dell'efficiente scalabilità elastica dell'infrastruttura cloud.
- 
**Resilienza:** Flussi di integrazione isolati vengono implementati in contenitori separati, che non possono influire gli uni sugli altri, sottraendo risorse condivise, come memoria, connessioni o CPU.







### Tecnologia: infrastruttura di integrazione cloud-native

L'integrazione agile richiede che le funzionalità di integrazione siano implementate utilizzando principi cloud-native. In questo modo, si abilita una piattaforma cloud-native, come ad esempio quella basata su Kubernetes, ad eseguire gran parte delle attività di implementazione, ridimensionamento e disponibilità in modo standardizzato. L'integrazione agile normalizza l'insieme di competenze di integrazione richieste per configurare e gestire una determinata funzionalità e garantisce la portabilità di tale funzionalità attraverso gli ambienti multicloud.

L'adozione di un approccio cloud-native ha un impatto sui modi in cui i team DevOps interagiranno con l'ambiente e la soluzione nel suo complesso per accrescere l'efficienza, man mano che un numero maggiore di soluzioni viene spostato su architetture portatili e leggere.

Inoltre l'approccio e la tecnologia cloud-native semplifica l'automazione della creazione e dell'implementazione di pipeline e consente un allestimento e una demolizione più puliti di ambienti di test. Questi vantaggi contribuiscono alla realizzazione di cicli di creazione-rilascio molto più brevi e ad un significativo incremento nella produttività.

Il passaggio ad un approccio cloud-native rappresenta un cambiamento significativo. Alcune caratteristiche di alto livello includono:

- 
**Componenti granulari:** Applicazioni che vengono suddivise in componenti di dimensioni inferiori, che possono essere modificati facilmente e ridimensionati in modo più indipendente e con minor rischio.
- 
**Runtime leggeri:** Componenti minimalisti consentono un'implementazione, un avvio e una scalabilità rapidi, per aiutare a garantire il miglior uso delle risorse dell'infrastruttura.
- 
**Infrastruttura come codice:** Un'infrastruttura creata direttamente a partire dalla sua descrizione codificata aiuta a garantire che le topologie siano ripetibili e portatili, per fornire ambienti attendibili ma anche flessibili.
- 
**Metodi agili:** Piccole distribuzioni regolari di funzionalità, con minimi impedimenti alla produzione, aiutano a garantire l'allineamento alle esigenze di business correnti.

Se si aderisce ai principi menzionati in precedenza, piattaforme di orchestrazione dei contenitori, quali Kubernetes, forniscono ridimensionamento automatico, bilanciamento del carico, implementazione, instradamento interno, reintegrazione e altro ancora, in modo standardizzato, per semplificare notevolmente l'amministrazione della piattaforma.

I moderni portafogli di integrazione devono essere in grado di adottare gli standard e i principi base di un approccio cloud-native. Funzionalità di runtime di integrazione, gestione API, flussi di eventi e messaggistica devono tutte utilizzare efficacemente piattaforme di orchestrazione dei contenitori per fornire la portabilità, la congruenza operativa e l'automazione richieste dalle soluzioni di oggi.

## Piattaforme di integrazione ibrida per l'integrazione agile

Una piattaforma di integrazione – o come alcuni analisti la chiamano “piattaforma di integrazione ibrida” – riunisce le funzionalità di integrazione di base in modo coesivo, così le organizzazioni possono creare integrazioni per le loro soluzioni di business in modo più efficiente e coerente.

Una piattaforma di integrazione ibrida è una raccolta di funzionalità di integrazione, in esecuzione sia on premise che sul cloud. Dovrebbe consentire a persone con livelli di competenze differenti (specialisti e non specialisti dell'integrazione) di eseguire un'ampia varietà di modelli di integrazione e implementarli separatamente per aumentare l'agilità.

Ai non specialisti dovrebbe essere fornita un'esperienza semplificata, che li guidi nella risoluzione di problemi chiari, mentre gli specialisti dell'integrazione sono esperti del controllo e possono occuparsi di modelli più complessi. In questo modo, si promuove la decentralizzazione, perché gli utenti possono lavorare in modo indipendente, ma collaborativo, con asset condivisi, pur mantenendo la governance.

La piattaforma di integrazione dovrebbe essere progettata in modo specifico per trarre vantaggio dall'infrastruttura cloud native, man mano che questa diventa di uso comune.

Leggi il più recente report di Ovum sui fattori che facilitano l'adozione della piattaforma di integrazione ibrida.

Scarica

IBM Cloud Pak  
for Integration –  
ottimizzato mediante  
tecnologia Intel.



## IBM Cloud Pak for Integration

**IBM Cloud Pak for Integration riunisce la principale serie di funzionalità di integrazione in una piattaforma coerente, che è semplice, veloce e affidabile.**

Adotta un approccio agile, accelerato dall'AI, all'integrazione, che consenta ai team estesi di creare integrazioni, utilizzare efficacemente una serie completa di stili e funzionalità di integrazione e incorporare AI e automazione nell'intero ciclo di vita dell'integrazione. Accelera lo sviluppo dell'integrazione, riduci i costi e incrementa l'efficienza operativa nel suo complesso, pur mantenendo un livello superiore di sicurezza, governance e disponibilità.

Crea API e integrazioni potenti in pochi minuti, fornisci scalabilità e prestazioni leader di mercato e offri funzionalità end-to-end senza eguali, con sicurezza di livello aziendale.

[IBM Cloud Pak for Integration](#) è la più completa piattaforma di integrazione ibrida nel settore e include tutti gli stili e le funzionalità di integrazione principali di cui il team ha bisogno. Distribuisce carichi di lavoro in qualsiasi ambiente on-premise o cloud per la massima flessibilità.

### **Integrazione di applicazioni e dati**

Da sistemi legacy complessi ad applicazioni SaaS semplici, connetti applicazioni e origini dati, on premise o nel cloud, per coordinare lo scambio di informazioni di business, in modo che i dati siano disponibili quando e dove sono necessari.

Connettori intelligenti precostituiti e un repository di asset condivisibile, comune, incrementano la velocità di distribuzione ed eliminano preoccupazioni relative alla mancata corrispondenza di origini, formati o standard.

Strumenti di integrazione a uso limitato di codice o senza codice utilizzano efficacemente NLP (natural language processing) e AI per offrire suggerimenti di associazione personalizzati in fase di creazione dei flussi di integrazione.

### **Gestione delle API**

Esponi e gestisci i servizi di business, come API riutilizzabili, per community di sviluppatori selezionate, sia interne che esterne all'organizzazione. Si può adottare una strategia API per accelerare la modalità di condivisione efficace degli asset di servizi e dati peculiari, per promuovere nuove applicazioni e nuove opportunità di business.

Utilizza efficacemente modelli AI pre-addestrati in base a chiamate API e dati di vendite o marketing, da origini quali CRM, per fare emergere ed evidenziare l'analytics del valore di business. Ridimensiona dinamicamente in base alla domanda, utilizzando procedure AIOps per rilevare, identificare e risolvere problemi di prestazioni della produzione e fornendo le informazioni di ritorno al processo di assemblaggio dell'integrazione, per addestrare la futura creazione di asset di integrazione basati su best practice.

## Messaggistica aziendale

Garantisci che le informazioni siano disponibili da qualsiasi ubicazione e in qualsiasi momento, tramite recapito affidabile dei messaggi, senza duplicazione o perdita di messaggi oppure operazioni complesse di ripristino nel caso in cui si verificano problemi di rete o di sistema. Failover completamente automatizzato, carichi di lavoro di messaggistica dinamicamente distribuiti, elevata velocità di trasmissione e suggerimenti per il riutilizzo delle code aiutano a fornire una soluzione ad alta disponibilità e bassa latenza.

## Flusso di eventi

Analizza dati associati ad un evento e rispondi a tali dati in tempo reale. L'azienda può accedere a dati inutilizzati, usufruire di insight sui dati in tempo reale e creare esperienze cliente reattive.

## Trasferimento dati ad alta velocità

Sposta enormi quantità di dati tra ambiente on premise e cloud o da cloud a cloud, rapidamente e in modo prevedibile, con livelli potenziati di sicurezza, per facilitare la rapida adozione di piattaforme cloud in caso di data set di dimensioni molto grandi.

## Sicurezza, automazione e monitoraggio a livello di piattaforma

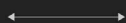
Estendi la connettività e l'integrazione oltre l'azienda con funzionalità perimetrali che proteggono le API, i dati che spostano e i sistemi di origine. Rileva problemi di produzione e agisci in base alle automazioni correlate.

### IBM Cloud Pak for Integration

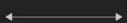
Esegui l'integrazione ad una velocità superiore del 300%, pur riducendo il costo dell'integrazione del 33%



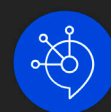
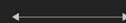
Gestione API



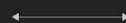
Integrazione di  
dati & applicazioni



Messaggistica  
aziendale



Flusso di eventi



Trasferimento ad  
alta velocità

**IBM Cloud Pak for Integration** è la piattaforma per l'integrazione agile. Con questa piattaforma, abbiamo visto aziende accelerare lo sviluppo della loro integrazione del 300%, ridurre i costi di integrazione di oltre il 33% e incrementare l'agilità operativa nel suo complesso.

Scopri di più