

Soluzioni per il cloud computing

I parte: Concetti basilari

Abstract: In questo articolo ci piace stare con la testa tra “le nuvole”. Partiamo quindi col descrivere cosa s’intende per cloud computing, quali sono le sue caratteristiche e i modelli di servizio che consente di fornire. In questo caso ci addentreremo nella descrizione di SaaS, PaaS e IaaS, sino alle tipologie di cloud pubblico, privato e ibrido.

tags: [Cloud Computing](#) [SaaS](#) [PaaS](#) [IaaS](#) [virtualizzazione](#) [pay-per-use](#) [data center](#)

Numero visite: 5358

Mi piace 854



di

Vittoria Caranna

Cloud Computing: come definirlo?

L’attenzione da parte della comunità ICT verso il Cloud Computing è alta grazie alla comparsa di un insieme di servizi, con caratteristiche comuni, proposti da importanti player di mercato.

Anche se alcuni concetti richiamati dal cloud come la **virtualizzazione** e l’**utility computing** non sono nuovi, la varietà di tecnologie che sono comprese nel cloud genera un po’ di confusione nel raffigurare un quadro complessivo. Con l’espressione **utility computing** ci si riferisce a un insieme di risorse che comprende capacità computazionale, storage e servizi. Questo concetto si rifà a quello di utility in senso più ampio, ovvero un servizio, come il gas, l’acqua e l’elettricità, erogati dietro un pagamento commisurato al consumo effettivo. Per questa ragione è stato anche la base sulla quale si è costruito il concetto di cloud computing e di SaaS. Va detto che l’enfasi che si sta dando attualmente a questo fenomeno, potrebbe addirittura compromettere la corretta interpretazione di uno scenario evolutivo. Spesso si ricorre all’hype cycle, originariamente proposto da Gartner che descrive la tendenza di nuove tecnologie a raccogliere un eccesso di interesse prima di raggiungere quella fase di maturità sufficiente per essere utilizzate dal mondo delle imprese. Un’evoluzione che parte da un entusiasmo eccessivo (**hype peak**), attraversa un periodo di disillusione (**through of disillusionment**) e può arrivare a una comprensione della rilevanza della tecnologia e di un suo ruolo nel mercato di riferimento. Il cloud computing è oggi posizionato nell’intorno dell’hype peak (figura 1).

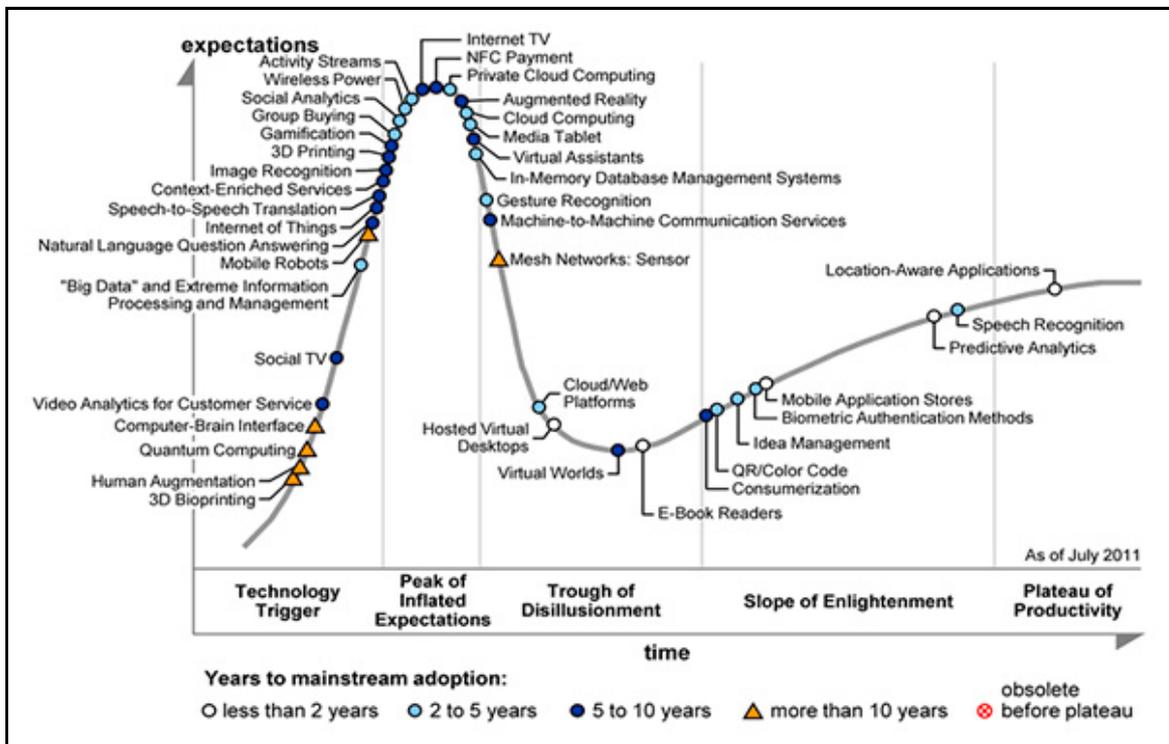


Figura 1 - Andamento evolutivo modelli e tecnologie secondo Gartner

Tutto quanto fa cloud computing

La confusione creata riguardo a questo nuovo paradigma e alle sue potenzialità, rende il cloud computing un'espressione **generalizzata** che include qualsiasi soluzione che consenta l'outsourcing di tutti i tipi di risorse informatiche (hosting e computing) e che richiama concetti quali l'**accesso trasparente** alle risorse secondo una logica **pay-per-use** e un'**infrastruttura** immediatamente scalabile **gestita da terze parti**. È per questo che è difficile trovare un'unica definizione di cloud computing; ne sono state proposte molte e tutte sembrano focalizzarsi solo su aspetti tecnologici.

Gartner identifica il cloud computing con uno **style of computing** in cui capacità IT scalabili sono fornite in modalità **as a service** via internet a una moltitudine di clienti esterni.

Per Forrester si fa riferimento a un **pool d'infrastruttura** gestita e altamente scalabile in grado di ospitare (hosting) applicazioni end-customer.

È interessante una ricerca compiuta da alcuni autori e pubblicata sull'ACM Computer Communication Review in cui sono state analizzate **più di 20 definizioni**, con la finalità di individuare le caratteristiche rilevanti del cloud computing, per tentare di raggrupparle in una completa definizione e trovare un comune denominatore. Alcuni autori sostengono che gli elementi chiave per il cloud siano la **scalabilità immediata** e l'**ottimizzazione dell'impiego** delle risorse disponibili grazie al monitoraggio e all'automatizzazione della gestione delle risorse in un ambiente dinamico. McFedries propone il **data center** (inteso come vasto insieme di cluster, ovvero gruppi di computer che lavorano insieme al fine di portare a termine un'elaborazione complessa come se fossero un unico dispositivo) quale elemento base del cloud per poter offrire elevate quantità di potenza elaborativa e storage impiegando risorse remote. Anche la **virtualizzazione** gioca un ruolo importante in quanto è stato definito da alcuni autori come l'insieme di hardware e software resi disponibili in maniera virtuale che hanno la capacità di operare contemporaneamente a diverse istanze pur in presenza di una singola macchina o di un solo sistema operativo.

Alla luce dell'analisi compiuta, Vaquero propone la seguente basilare definizione, con la consapevolezza che ci si sta riferendo a un concetto in continua evoluzione:

"Il cloud computing rappresenta un vasto insieme di risorse virtualizzate, facilmente accessibili e usabili. Queste risorse possono essere dinamicamente riconfigurate per adattarsi a un carico variabile: ciò consente anche un loro ottimale utilizzo. Questo insieme di risorse è tipicamente offerto con un modello pay-per-use le cui garanzie sono fornite dall'Infrastructure Provider sulla

base di concordati SLA."

Le caratteristiche che possono richiamare un comune denominatore per il cloud sono quindi la **scalabilità**, il modello **pay-per-use** e la **virtualizzazione**.

Caratteristiche del Cloud Computing

Gli elementi che caratterizzano una soluzione cloud possono essere quelli riportati di seguito, come suggerito da Rhoton

Servizio on-demand

Il servizio è in host e fornito da una postazione appartenente a un service provider. Ciò implica che il servizio è erogato attraverso Internet e che l'elaborazione avviene al di fuori dell'azienda.

Ampia rete di accesso

Le risorse sono disponibili in rete e accedibili tramite meccanismi standard messi a disposizione da varie piattaforme client come laptop, telefonia mobile e PDA.

Pool di risorse

I provider delle risorse sono organizzati per riuscire a rispondere a più utenti utilizzando il modello multi-tenant, con varie risorse fisiche e virtuali assegnate dinamicamente in base al numero di richieste che giungono. Il senso è quello di astrazione della locazione dal momento che l'utente non ha controllo, e addirittura non ha conoscenza dell'esatta locazione del provider.

Elasticità rapida

Uno dei vantaggi offerti dal cloud computing riguarda l'elasticità del fornitore del servizio nei confronti dell'utente finale. Il modello di fornitura si basa su un meccanismo molto flessibile che consente di quantificare in modo molto rapido le risorse richieste con effetti positivi sul risparmio dei costi.

Virtualizzazione

I servizi cloud sono generalmente offerti attraverso un'infrastruttura astratta. Sono definiti secondo diversi livelli di virtualizzazione e con la multi-tenancy se ne possono ottimizzare i costi.

Multi-tenancy

Il cloud è usato da molte aziende (**tenant**) e il provider deve comprendere meccanismi tali a proteggere e isolare ogni tenant rispetto agli altri. La condivisione di risorse consente sia la scalabilità sia il risparmio di costi e non compromette la privacy e la sicurezza dei dati di ciascuna impresa.

Servizi di monitoraggio

I sistemi cloud sono predisposti per poter controllare e ottimizzare le risorse usate, mitigando l'utilizzo delle loro potenzialità sulla base del servizio erogato (storage, larghezza di banda, account di utenti attivi). Quindi le risorse possono essere monitorate, controllate con una garanzia di trasparenza sia per il provider e consumer del servizio utilizzato.

Non è necessario che ogni elemento sopraindicato sia assolutamente indispensabile per il cloud computing e non si vuole neanche dire che questi elementi sono prerequisiti necessari e sufficienti per una nozione di cloud. Tuttavia tanto più essi sono presenti, tanto maggiore è la probabilità di essere di fronte a una soluzione cloud.

Gli attori, gli scenari e i modelli di servizio

I principali **attori** del cloud sono: i Service Provider (**SP**), i Service User (**SU**) e gli Infrastructure Provider (**IP**) (figura 2).

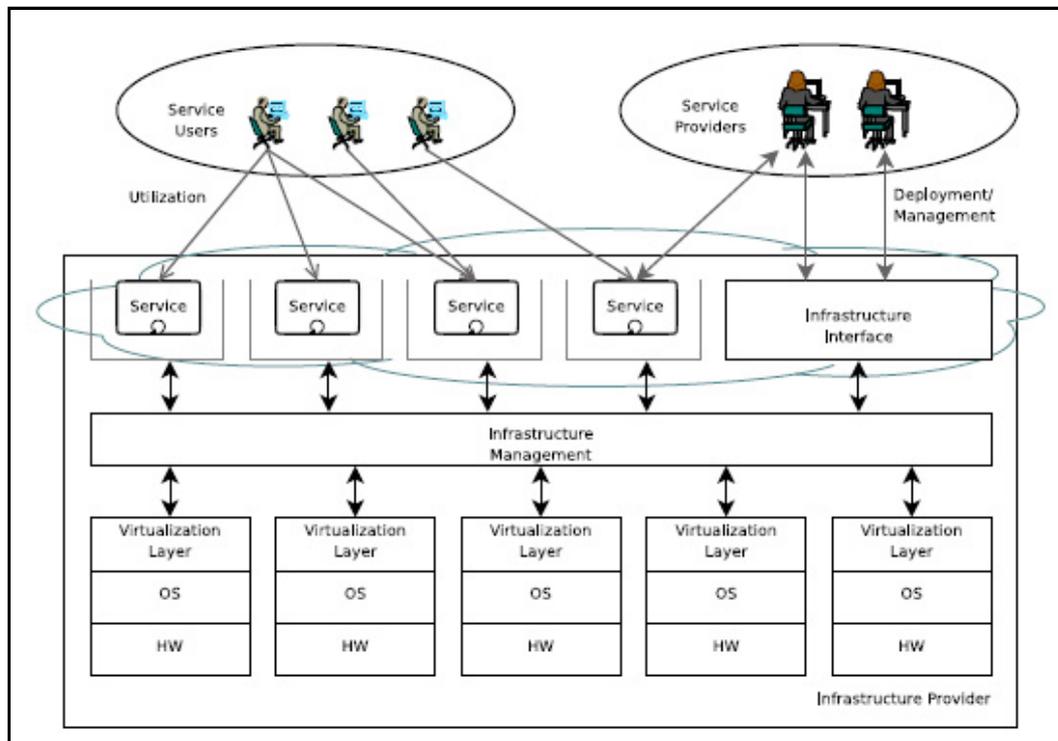


Figura 2 - Relazione tra Service Provider, Service User e Infrastructure Provider in ambiente cloud.

I **Service Provider** rendono i servizi accessibili ai **Service User** attraverso interfacce Internet-based. Il cloud ha lo scopo di esternalizzare la fornitura dell'infrastruttura richiesta ai servizi host: questa infrastruttura è offerta come "servizio" dagli **Infrastructure Provider**.

Secondo Marinos e Briscoe invece gli attori sono i **vendor**, gli **sviluppatori** e gli **utenti finali**. In tale caso, il vendor ha il ruolo di fornire le risorse, gli sviluppatori utilizzano le risorse fornite per creare servizi usufruiti dagli utenti finali. Spesso i ruoli però possono essere molteplici cioè vendor che sviluppano anche servizi direttamente per gli utenti finali o sviluppatori che utilizzano i servizi di altri per creare servizi propri.

Gli **scenari** possibili sono il **SaaS** (Software as a Service), il **PaaS** (Platform as a Service) e lo **IaaS** (Infrastructure as a Service) che saranno approfonditi nel paragrafo successivo come modelli di servizio (figura 3).

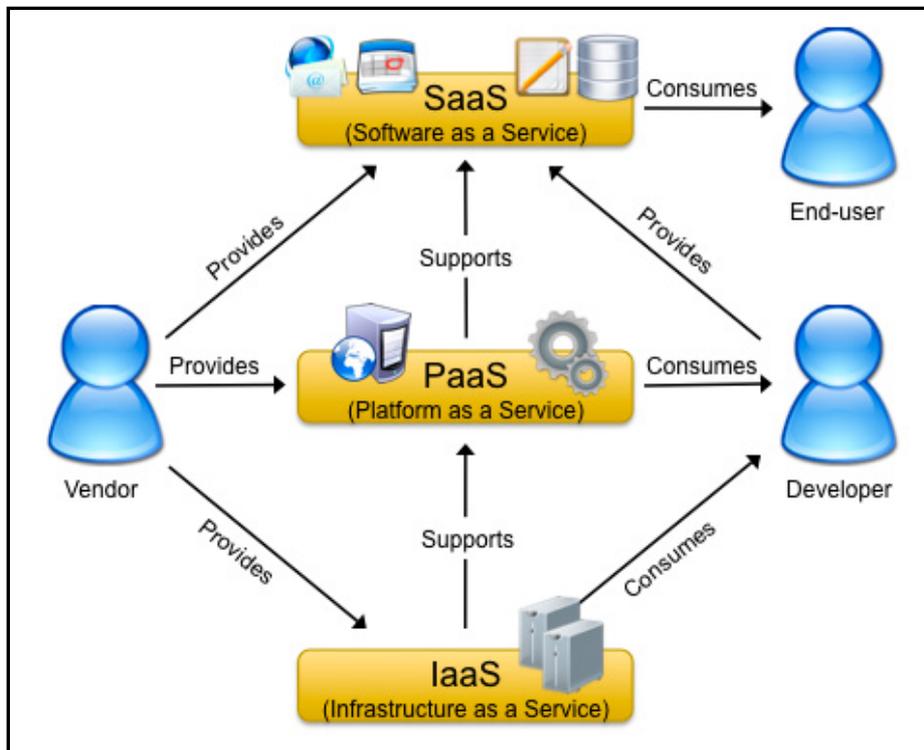


Figura 3 - Relazione tra SaaS, PaaS, IaaS e Vendor e Developer.

Come mostra la figura 3, si può notare la relazione (verticale) che esiste tra i tre modelli di servizio, nel senso che lo IaaS supporta il PaaS che a sua volta fa da supporto al SaaS finale.

Modelli di servizio

Vediamo adesso quali sono i **modelli** messi a disposizione dal cloud secondo i quali è possibile erogare dei servizi.

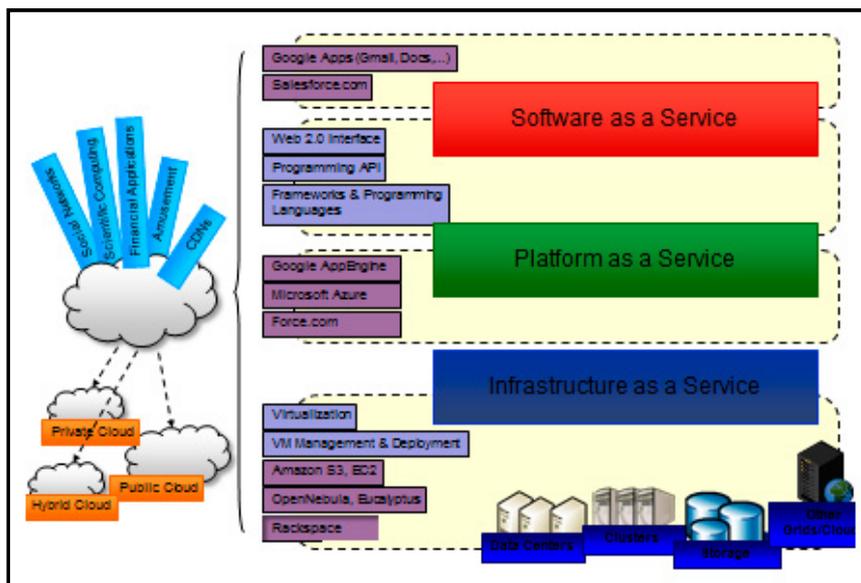


Figura 4 - Relazione tra IaaS, PaaS, SaaS.

Come mostra la figura 4, è importante chiarire sin da subito la relazione che esiste tra i tre modelli di servizio nel senso che si comportano come delle scatole cinesi. Fornire il SaaS, che è il servizio di più alto livello che può essere offerto ed è quello per l'utente finale, significa offrire anche le componenti tipiche dello IaaS e del PaaS; offrendo il PaaS, si offrono anche tutte le componenti previste nello IaaS. Con quest'ultimo invece si offre il livello base previsto dai modelli cloud.

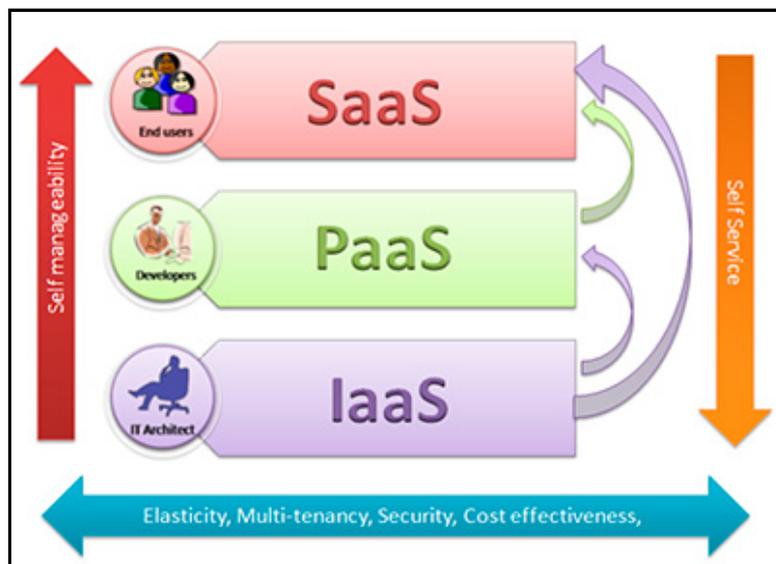


Figura 5 - Il rapporto tra caratteristiche del cloud e tipologia di utilizzatori, in base alla gestibilità delle soluzioni, e al livello del servizio fornito.

Nei prossimi paragrafi analizzeremo caratteristiche, vantaggi e svantaggi dei diversi modelli di servizio.

Cloud di tipo Software as a Service (SaaS)

Il **SaaS** è un modello per la distribuzione di applicazioni in cui il fornitore (che può anche essere lo sviluppatore dell'applicazione web) eroga l'applicazione direttamente tramite la rete Internet, consentendone così l'utilizzo agli utenti. A differenza di quanto accade con le tradizionali applicazioni a package che l'utente prima acquista, poi installa sui propri server e quindi esegue in locale, nelle applicazioni SaaS il software è di proprietà del provider, il quale ospita **sia l'applicazione sia i dati**, fornisce ai suoi clienti accesso alle applicazioni, alle patch e agli aggiornamenti necessari in modo trasparente, tramite uno smart client o un'altra applicazione.

I clienti non pagano per il possesso del software, ma solamente per il suo utilizzo e quindi per il valore reale che esso genera. Il provider dispone di un'**architettura centralizzata**, detta **multi-tenant**, che permette a più utenti di condividere una piattaforma comune e di usufruire simultaneamente dello stesso servizio; la personalizzazione è comunque garantita grazie all'utilizzo di semplici configurazioni dei metadati, che permettono di descrivere le preferenze dei diversi utenti.

I **vantaggi** conseguibili grazie all'utilizzo di questo paradigma vanno dalla riduzione del costo dell'infrastruttura, in quanto questa viene condivisa, al conseguimento di economie di scala per il provider, alla rapidità del deploy, dal momento che non è necessario installare alcun software in house e infine alla possibilità di utilizzare sempre la versione più aggiornata dell'applicazione.

Le **caratteristiche distintive** principali del software come servizio sono l'ubicazione del codice applicativo, la modalità di deployment e di accesso. Ma secondo questa definizione, il software come servizio sarebbe semplicemente un software distribuito come servizio in hosting a cui si accede tramite Internet.

Per passare dall'offerta del SaaS come servizio di hosting e di software sviluppato internamente all'offerta di software come servizio, i fornitori devono concepire in modo nuovo tre elementi intercorrelati: il modello di business, l'architettura dell'applicazione e la struttura operativa. Vediamo cosa vogliamo intendere.

Modello di business

Avere un **modello di business** aderente al SaaS significa intervenire in:

- **Riallocazione della responsabilità** della gestione e dell'infrastruttura tecnologica (l'hardware e i servizi professionali passano infatti dal cliente al provider).
- **Riduzione del costo** della fornitura dei servizi software, per effetto della specializzazione e dell'economia di scala.
- **Acquisizione di una clientela** di aziende minori, attraverso la riduzione del costo minimo a cui il software può essere venduto.

- Scelta del **giusto approccio**. Esso può essere **isolato** o **condiviso**. **Isolato** significa offrire la propria applicazione a costi più alti. In alcuni casi questa scelta può essere tuttavia giustificata dalla necessità di soddisfare esigenze diverse. **Condiviso** significa eseguire una singola istanza per tutti i clienti e assicurare la possibilità di configurare l'aspetto e le funzionalità dell'applicazione tramite metadati. Con questo approccio non è più necessario che il server del fornitore disponga di spazio sufficiente a contenere un numero di istanze equivalente al numero di clienti. Le risorse di elaborazione vengono utilizzate in modo molto più efficiente e i costi vengono ridotti di conseguenza. Per consentire a posteriori la scalabilità orizzontale, occorre strutturare il database condiviso tra più tenant (p.e. aggiungendo id tenant) di modo che il partizionamento sia basato sull'Identificativo di quest'ultimo, così da poter indirizzare i dati tenant individuali e spostarli per esempio su un altro server.

Modello architetturale

Avere un **modello architetturale** aderente al SaaS significa che un'applicazione SaaS deve essere:

- **Scalabile**: rendere **scalabile** un'applicazione significa aumentare la concorrenza massima e utilizzare le risorse applicative in modo più efficiente, condividendo connessioni di rete, memorizzando dati di riferimento nella cache e dividendo i database di grandi dimensioni in più parti.
- **Multi-tenant** (multi organizzazione): il multi-tenancy è la novità più significativa ed è pertanto necessario che l'architettura preveda la massima condivisione delle risorse tra i diversi utenti e il necessario isolamento tra i dati di ciascuno.
- **Configurabile**: naturalmente, se una singola istanza di applicazione in esecuzione su un singolo server deve accogliere utenti di diverse aziende contemporaneamente, non è possibile limitarsi a implementare codice personalizzato per rispondere alle esigenze di un utente specifico, perché ogni modifica impatterebbe anche su tutti gli altri utenti. Anziché accedere a una versione personalizzata dell'applicazione in senso tradizionale, ogni cliente utilizzerà **metadati per configurare** il funzionamento e l'aspetto della stessa applicazione.

Modello operativo

Avere un **modello operativo** aderente al SaaS significa:

- Garantire il rispetto del **contratto di servizio** (SLA) senza ricorrere all'isolamento.
- Esaminare attentamente gli **obblighi previsti dai contratti** di servizio sottoscritti con i clienti, con particolare attenzione alle clausole relative ai tempi di inattività, alle opzioni di supporto e al ripristino d'emergenza e determinare se tali obblighi possono essere soddisfatti con un'architettura applicativa in cui più clienti non correlati condividono l'accesso a una stessa istanza di applicazione.

Svantaggi nell'utilizzo del SaaS

Come qualsiasi tecnologia o modello di business, ci sono anche degli **svantaggi** in quanto **le tecnologie ed il relativo business model dietro ai servizi cloud non hanno ancora raggiunto la maturità**. I reparti di ricerca e sviluppo dei provider stanno ancora lavorando attivamente per dare ai clienti il livello di servizio e la qualità richiesta dai processi aziendali più critici.

Vediamo di seguito quali sono i principali svantaggi nell'adozione di una soluzione SaaS

Sicurezza

Ecco alcuni degli inconvenienti relativi alla sicurezza da considerare quando si ricorre a una soluzione SaaS. **Non** è possibile avere il **pieno controllo dei dati** aziendali in quanto non si ha il controllo dei server a cui sono stati consegnati.

La **cancellazione di dati non implica** l'immediata **cancellazione fisica** e, soprattutto, **sicura** (ad esempio con algoritmi approvati dal DoD) degli stessi all'interno dell'infrastruttura cloud.

Inoltre, al momento **non** ci sono **standard** di sicurezza **ratificati** per i sistemi cloud.

La collocazione fisica dell'hardware e del software è sconosciuta: **ispezioni, procedure di audit, hardening e intrusion test sono difficoltosi** da eseguire.

Resta sempre il rischio di perdita dei dati causati da rotture hardware o problemi a uno degli strati di astrazione software.

Va ricordato che, almeno per quanto riguarda USA e Unione Europea, i provider più sicuri devono aderire alla **Safe Harbor Privacy Principles** che consente il trasferimento dei dati verso Paesi terzi solo se lo Stato di destinazione presenta un livello di protezione "adeguata" dei dati.

Dipendenza (perdita di controllo)

Per quanto attiene alla **dipendenza** che si acquisisce, possiamo parlare di una **perdita di controllo** rispetto a una soluzione in house. Infatti non si ha nessuna influenza sulla qualità, la frequenza e sulle tempistiche degli **interventi di manutenzione** operate dal provider.

Vi è poi il rischio di incorrere in nessuna o poca conoscenza in merito alle procedure operate dal provider (esecuzione di backup, ripristino dei dati e procedure di disaster recovery).

Da non dimenticare è anche la difficoltà di **migrazione verso un altro provider** in quanto non ci sono standard che uniformino il mercato; e questo problema potrebbe esplodere soprattutto tra qualche anno.

La misurazione delle risorse utilizzate e delle attività utente è completamente nelle mani del provider.

Perdita di conformità

Se l'adesione a standard di conformità è importante (e in genere lo è), occorre considerare che la soluzione SaaS implica quanto segue:

- impossibilità di mantenere **certificazioni** acquisite;
- indisponibilità del provider per **audit** di terze parti;
- indisponibilità del provider a acquisire certificazioni;
- mancanza di requisiti per specifiche certificazioni (p.e. **PCI DSS**).

Flessibilità

Altro aspetto da tenere presente è quello della flessibilità. Con le soluzioni SaaS, è **impossibile** richiedere **personalizzazioni dell'infrastruttura**. Inoltre, l'implementazione di **tecnologie innovative** è dettata dal provider che potrebbe rallentare l'adozione a seconda delle decisioni prese dalla sua governance aziendale.

Costi

Sebbene quello della riduzione dei costi sia un indubbio vantaggio del cloud computing, occorre però valutare attentamente alcuni costi aggiuntivi che potrebbero eventualmente presentarsi: si tratta di **costi indiretti**, ad esempio quelli legati al rispetto delle regolamentazioni/leggi oppure alla gestione dei backup e dei piani di disaster recovery. Oltre a ciò, c'è il rischio che i **costi dell'infrastruttura** aumentino imprevedibilmente in futuro: questo è uno svantaggio nel momento in cui la migrazione verso un altro provider non è affatto trasparente.

Conoscenze

Sotto questa voce vogliamo far notare che da un lato sono richieste **conoscenze maggiori** (legate anche agli aspetti legali e giurisprudenziali) per implementare e **gestire i contratti** SLA con il provider, mentre dell'altro si viene a perdere la conoscenza dell'infrastruttura, poiché essa (almeno nella sua completezza) è concentrata lato provider, ed è quindi impossibile controllare come agisce il provider.

Integrazione

Alcuni inconvenienti legati all'integrazioni sono i seguenti. L'integrazione con attrezzature presenti in altri data center è difficile da portare a termine. Inoltre, l'integrazione con le **periferiche "locali"** (come le stampanti o altri dispositivi USB) e con gli **apparati di sicurezza** è complessa (e in alcuni casi impossibile).

Vantaggi del paradigma SaaS

Il Cloud Computing, se sfruttato in maniera opportuna, può dare significativi benefici, a patto che si tengano ben presente alcuni punti:

- Bisogna **comprendere i rischi** cui si va incontro, come quelli citati sopra relativi alla sicurezza, alla perdita di controllo e di conformità, alla flessibilità, ai costi, alle conoscenze e all'integrazione.
- L'utilizzo di una infrastruttura cloud richiede **modifiche nell'organizzazione aziendale e/o del reparto IT**: cambio di procedure, richiesta di nuove conoscenze, costi fisici e organizzativi, etc.
- È fondamentale **comprendere** con estrema attenzione il **contratto** di fornitura dei servizi e le eventuali **SLA** annesse.
- È caldamente consigliato formulare e mantenere aggiornata una **exit strategy** nel caso in cui si renda necessario

cambiare provider.

- Si deve cercare di scegliere il provider che implementa **quanti più standard open** siano possibili.

Un esempio molto famoso di SaaS sono i **documenti Google** ora disponibili in **Google Drive**, una suite per ufficio che comprende un word processor, un foglio di calcolo, un'applicazione per le presentazioni e una per disegnare. In questo caso, il cloud provider (cioè Google) ha scelto di rendere il suo SaaS fruibile a chiunque, senza alcun costo.

Cloud di tipo Platform as a Service (PaaS)

Un malinteso comune tra gli sviluppatori è che il cloud computing si applica e viene incontro solo alle esigenze dei sistemisti di rete. Invece non è così: quante volte è risultata una scocciatura e un processo ripetitivo dover configurare e preparare l'ambiente di sviluppo che prevede l'installazione e la configurazione di un web server in cui fare deploy della nostra applicazione web? Il PaaS vuol rispondere proprio a questo tipo di esigenza, fornendo una piattaforma pronta, sui cui installare quanto serve.

Ragioni per l'utilizzo del PaaS

Quando il PaaS non è sfruttato, preparare un ambiente di sviluppo prevede questi macropassi:

- acquisizione del **server** per il deploy;
- installazione dell'ambiente runtime (**ambiente di esecuzione** per applicazioni), del **repository** per il codice sorgente e dei **middleware** necessari;
- import del **codice software** esistente nell'ambiente e/o sviluppo dello stesso;
- **testing** ed **esecuzione** del programma.

Forse il modo migliore per capire il PaaS è quello di dedicarsi ai suoi principali componenti: la **piattaforma** e il **servizio**. Per illustrare questi due "ingredienti", cerchiamo di approfondire le loro definizioni. Una **piattaforma di elaborazione**, nella sua forma più semplice, si riferisce al posto in cui il software può essere eseguito. Esempi comuni di piattaforme sono, nelle loro varie versioni, Microsoft Windows, Apple Mac OS X e Linux per i sistemi operativi desktop; Android, Windows Mobile e Apple iOS per il mobile computing, oppure Adobe AIR o Microsoft NET Framework. La cosa importante è che non si sta parlando del software stesso, ma della piattaforma sulla quale essa deve funzionare. La figura 6 dà una rappresentazione per aiutare a capire questo tipo di relazione.

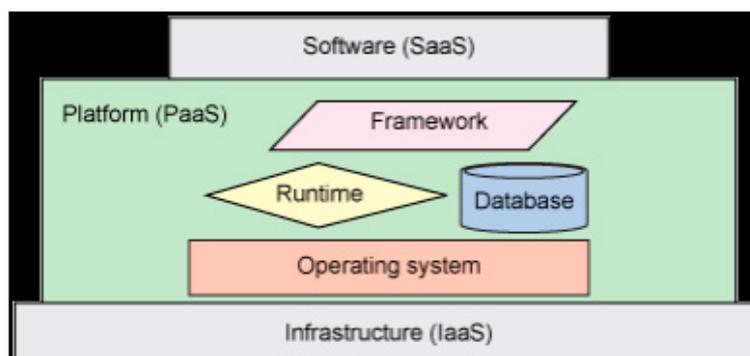


Figura 6 - Lo stack di tecnologie necessarie al funzionamento di applicazioni: alla base c'è l'infrastruttura e, via via salendo, si arriva fino alle applicazioni vere e proprie. Notare come la piattaforma di elaborazione inglobi diversi componenti.

Una volta precisato cosa si intende per piattaforma di elaborazione, proviamo a chiarire cos'è invece uno stack di soluzione dal momento che è fornito come servizio: consiste nell'insieme di quelle applicazioni che assistono nel processo di sviluppo e si riferiscono al sistema operativo, agli ambienti runtime, ai repository di versionamento e a qualsiasi tipo di middleware.

Scelta del fornitore

Lo stack di soluzione viene proposto da molti fornitori ed è una cosa da valutare nel momento in cui si decide di saltare sul treno del PaaS. Ecco alcune considerazioni e le domande che occorre porsi per poter scegliere un fornitore di PaaS:

- Che **framework** e che **tipo di linguaggi** supporta? Idealmente, un PaaS dovrebbe supportare ogni framework di linguaggio della piattaforma scelta.

- **Quante applicazioni** è possibile creare? La maggior parte di fornitori di PaaS limitano il numero delle applicazioni realizzabili, sulla base del tipo di offerta per la quale ci si è registrati. Occorre quindi assicurarsi che il fornitore offra un piano che soddisfi le esigenze.
- Verificare che l'infrastruttura supportante il multi-tenant metta a disposizione una **VM dedicata** a ciascuna organizzazione e gestita dall'hypervisor.
- Che **tipo di database** sono supportati? Questo è rilevante se i propri dati devono interagire con varie sorgenti dati e occorre quindi verificare che i database offerti siano compatibili col formato dati delle proprie applicazioni.
- Verificare che sia supportato l'HTTPS, specialmente se le proprie applicazioni coinvolgono delle transazioni.

Svantaggi nell'utilizzo del PaaS

Per quanto riguarda gli svantaggi, oltre a quelli evidenziati per il SaaS, si aggiungono i seguenti problemi:

- **Portabilità tra differenti provider PaaS.** Ogni cloud provider usa implementazioni proprie (p.e.: l'interfaccia delle strutture dati come la hash table) che rendono molto difficile il "trasloco" di un software da un provider all'altro, malgrado il linguaggio di programmazione dei due PaaS possa essere lo stesso. Come rimedio, gli sviluppatori possono cercare di programmare per interfacce (che è in generale una buona pratica di programmazione), ma ciò può andare a scapito dell'efficienza, perché non permette di sfruttare appieno le possibilità offerte da uno specifico provider.
- **Complessità di un'applicazione PaaS.** A differenza di un'applicazione che deve essere eseguita in un ambiente locale e isolato, un'applicazione PaaS è molto più complessa, in primo luogo perché deve accedere alla rete. Perciò, lo sviluppatore deve considerare molteplici aspetti concernenti la sicurezza delle trasmissioni (la crittografia su tutti), e deve padroneggiare molte tecnologie e linguaggi di programmazione (HTML, HTTP, XML, etc.).

Google App Engine è l'offerta PaaS di Google, una piattaforma che permette lo sviluppo di applicazioni web e l'hosting delle stesse presso i data center di Google. Tali applicazioni possono così beneficiare di un alto grado di scalabilità, lo stesso di cui gode ognuna delle Google Apps. Al momento, i linguaggi di programmazione supportati sono **Java**, **Python** e **Go**. Per ognuno di questi linguaggi, Google ha pubblicato dei **Software Development Kit (SDK)** ad hoc, che si possono scaricare liberamente dal sito e installare sulla propria macchina per cominciare a sviluppare. L'utilizzo della piattaforma di Google è gratuito, almeno inizialmente, ed è possibile poi acquistare delle ulteriori risorse computazionali, di memoria e di banda, in funzione delle esigenze.

Cloud di tipo Infrastructure as a Service (IaaS)

Gli **IaaS** sono i servizi che permettono di avere l'accesso al **livello più basso di cloud**. Un'azienda fornitrice di IaaS mette a disposizione una infrastruttura hardware remota e ci fornisce la documentazione necessaria affinché la nostra piattaforma software possa utilizzarne le risorse (cicli CPU, memoria RAM, storage). Volendo fornire un esempio concreto, un provider che mette a disposizione la propria infrastruttura, per l'esecuzione di virtual machines e per la loro gestione con assegnazione di risorse on demand, è un **IaaS Provider**. Alcuni dei concetti chiave della IaaS includono:

- **Cloudbursting:** è l'esigenza di poter scalare velocemente le proprie applicazioni di rete per soddisfare improvvisi picchi di domanda.
- **Multi-tenant:** abbiamo già visto questo concetto che è l'utilizzo da parte di più organizzazioni.
- **Pool di risorse** ossia un subset delle risorse fisiche per gestire uno o più gruppi di carico di lavoro.
- **Hypervisor:** si tratta di un Virtual Machine Manager, cioè di un vero e proprio gestore delle macchine virtuali, che è il componente centrale e più importante di un sistema basato sulle VM.

La cosa più importante è conoscere i due aspetti principali che rendono IaaS speciale: **elasticità** (la capacità di ridimensionare velocemente le dimensioni di storage, memoria, etc.) e **virtualizzazione** (tecnologia emergente che permette a un server fisico ospitante di gestire sistemi operativi, andando a emulare le istanze delle macchine non più fisiche ma, appunto, virtuali).

Il valore dello IaaS

Per le imprese, il valore aggiunto dello IaaS è dato dal concetto di **cloudbursting**, cioè l'attività di caricare più task nel cloud nei periodi in cui sono necessarie più risorse di calcolo. Il potenziale di risparmio dei capitali attraverso il cloudbursting è significativo, perché le imprese non avranno bisogno di investire in server aggiuntivi che funzionano solo al 70% della capacità per due o tre volte nel corso dell'anno, mentre per il resto del tempo sarebbero sfruttati solo per il 7-10% delle loro potenzialità.

Certo, le organizzazioni devono fare attenzione allo stato attuale del mercato di cloud computing in termini di vendor lock-in. Le macchine virtuali (VM) possono essere spostate dai data center al cloud ma soprattutto nelle nuvole di diversi fornitori, ma questa operazione richiede che i fornitori supportino un formato standard del software che deve essere ospitato.

La realtà dei fatti è che **attualmente non vi è alcuna indicazione** a riguardo il che complica le cose, perché non vi è alcuna garanzia che il formato attorno al quale si costruisce oggi sarà un domani sostenuto nel tempo. Va notato, tuttavia, che spesso è però possibile portare una macchina virtuale in un altro formato, a condizione che la specifica del nuovo formato sia aperta o che si possa avere accesso ad esso. Una nota promettente è che sono stati compiuti grandi progressi a sostegno dell'**Open Virtualization Format (OVF)**, candidato promettente per diventare uno standard. Un altro candidato promettente è il formato **Virtual Machine Disk (VMDK)**, in origine proprietario per VMware, ma che ora la specifica ha aperto e che quindi è supportato da un numero più ampio di fornitori.

La prima sfaccettatura dello IaaS: l'elasticità

Abbiamo detto che l'**elasticità** è il primo aspetto vantaggioso dello IaaS. Immaginiamo che le nuvole siano degli ammassi aggregati di cluster di caramelle di modo che la gente ci si possa sedere sopra. Ogni nuvola di caramelle può ospitare solo un certo numero di persone, a seconda del numero di cluster che compongono la nuvola e dal numero di caramelle contenute in questi cluster. È possibile espandere i cluster di caramelle semplicemente aggiungendone altre. Come probabilmente avete già capito, le persone rappresentano le applicazioni che richiedono risorse di calcolo, come ad esempio quelle che ospitano siti Web. I cluster di caramelle rappresentano gruppi di macchine virtuali, in cui ogni caramella è una VM. Questo concetto è noto come **clustering elastico**: la capacità cioè di ridimensionare le capacità computazionali nel cloud. I principali fornitori di cloud danno la possibilità di scalare rapidamente le risorse messe a disposizione nel cloud.

La virtualizzazione delle macchine

Con lo IaaS si ha la capacità di fornire elaborazione, storage, reti e altre risorse di calcolo, in cui è possibile fare deploy ed eseguire software come sistemi operativi e applicazioni. La **virtualizzazione** rende il paradigma cloud unico, nel senso che realizzare uno IaaS significa essere indipendenti dalla piattaforma e fare una combinazione di risorse hardware e software. Per essere indipendenti dalle piattaforme in uso, si può utilizzare un **hypervisor** che è responsabile di gestire più macchine virtuali ospitanti sistemi operativi differenti (figura 7). Questo processo viene indicato come pool di risorse che rende possibile la virtualizzazione e di conseguenza la gestione del multi-tenant, concetto che si riferisce ad una infrastruttura condivisa da più organizzazioni con esigenze comuni.

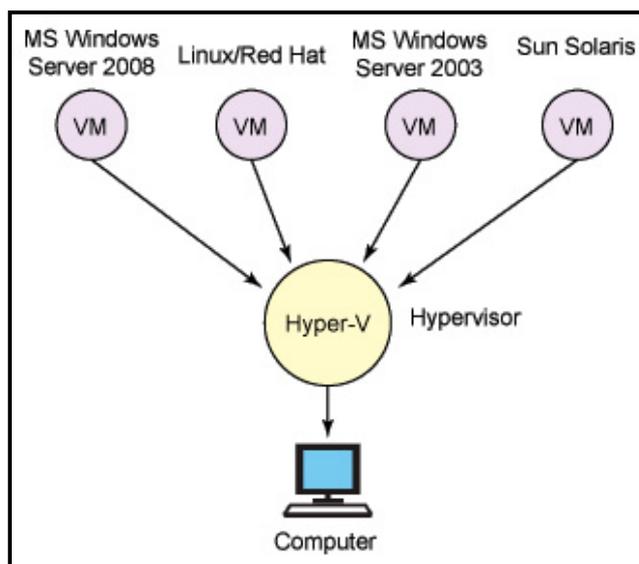


Figura 7 - Hypervisor.

Svantaggi nell'utilizzo dello IaaS

Tra gli svantaggi della Infrastructure as a Service citiamo invece:

- l'uso della **rete** e le problematiche associate, quali la sicurezza delle comunicazioni e la grande dipendenza dalla connessione di rete (disponibilità e prestazioni, non sempre scontate);
- se sulle VM si eseguono delle applicazioni **legacy**, ne possono derivare dei problemi di sicurezza dovuti alle vulnerabilità di quelle stesse applicazioni.

Tipi di cloud

Oltre che da un punto di vista "tecnologico", è possibile classificare i sistemi di cloud computing in base al loro grado di "proprietà" da parte di un provider e degli utenti: **pubblico**, **privato** e **ibrido**. Vediamo adesso sotto quali tipologie si può presentare la "nuvola" e in quali contesti è conveniente utilizzarne una piuttosto che un'altra.

Cloud pubblico

Un **cloud pubblico** si basa sul modello standard di cloud computing, nel quale un service provider rende disponibili al pubblico su Internet le risorse, come applicazioni e storage. I servizi di cloud pubblico possono essere gratuiti o offerti secondo un modello pay-per-usage. I principali **vantaggi** dell'utilizzare servizi di **public cloud** sono:

- **set-up semplice** ed economico perché i costi dell'hardware, delle applicazioni e della banda sono coperti dal provider;
- **scalabilità** per soddisfare le esigenze degli utenti;
- **nessuna risorsa sprecata** perché si paga per quello che si usa.

I principali provider hanno aderito alle varie normative per il trattamento dei dati (come ad esempio la normativa "Safe Harbor" che consente di trasferire i dati dall'UE agli Stati Uniti) e gli svantaggi di un cloud pubblico sono appunto correlati alla sicurezza e privacy dei dati distribuiti su tali server al di fuori dei confini nazionali, requisito per esempio richiesto dalla normativa a molte grandi società italiane.

Altro aspetto da non sottovalutare nel rapporto con gli operatori di cloud pubblico è comprendere bene le modalità di recupero dei propri dati e di migrazione, nel caso si voglia cambiare operatore o tornare indietro con infrastrutture "ospitate in casa".

Cloud privato

Il **cloud privato** è un'architettura proprietaria che fornisce servizi hosted a un numero limitato di utenti. Gli sviluppi nella virtualizzazione e negli ambienti distribuiti hanno permesso agli amministratori delle reti aziendali e dei datacenter di diventare service provider efficienti che soddisfano le esigenze dei loro "clienti". Il cloud privato ha il punto di forza di poter essere adottato da un'azienda che desidera o ha bisogno di **maggior controllo** sui propri dati rispetto a quanto farebbe utilizzando un servizio di terze parti.

I principali **svantaggi** di un **private cloud** sono:

- il **mancato abbassamento del TCO** (Total Cost of Ownership) cioè non usufruire del risparmio sull'acquisto, installazione, manutenzione e dismissione di hardware e software;
- la **mancata focalizzazione esclusiva sul proprio core business**: è necessario assumere personale specializzato che si occupi della gestione dell'infrastruttura tecnologica, mentre con altre soluzioni ci si può concentrare in maniera esclusiva sui compiti e sugli obiettivi fondamentali dell'azienda;
- il rischio di ricadere nella **solita virtualizzazione**: questo aspetto è un supporto tecnologico alla realizzazione di un cloud privato, ma non occorre fermarsi alla realizzazione dell'infrastruttura, occorre dedicarsi all'esecuzione di un cloud privato che comprende il modo di organizzare, gestire e fornire i servizi IT.

D'altro canto, implementare una struttura basata su un cloud privato significa soddisfare cinque caratteristiche essenziali:

- **servizio su richiesta**: significa che ci deve essere una forma di portale con catalogo di servizi in grado di attivare automaticamente i flussi di lavoro per l'approvazione e il provisioning automatizzato;
- **ampio accesso di rete**: il servizio è accessibile tramite rete Internet o aziendale utilizzando vari dispositivi come laptop e dispositivi mobili;
- **pool di risorse**: implica la condivisione in tempo delle risorse IT (elaborazione, storage, rete) attraverso varie tecnologie abilitanti come la virtualizzazione;
- **flessibilità rapida**: capacità di aumentare o diminuire dinamicamente le risorse IT, a seconda delle esigenze mutevoli e, ove possibile, automaticamente;
- **servizio monitorato**: significa che la capacità e l'utilizzo del servizio viene monitorato di modo da tornare utile per la pianificazione delle previsioni future.

Cloud ibrido

Il terzo modello è il cosiddetto **cloud ibrido** che unisce il cloud privato e quello pubblico per cercare di ottenere il massimo dall'impiego dei punti di forza di entrambe le soluzioni. L'utilizzo di un'infrastruttura ibrida garantisce vantaggi in termini di costi e scalabilità e, allo stesso tempo, offre alle aziende i canoni di sicurezza di un cloud privato. Il cloud ibrido rappresenta la soluzione ottimale per quelle aziende che dispongono di un'infrastruttura consolidata e che vogliono migrare gradatamente verso un cloud pubblico, oppure per quelle aziende che vogliono delegare la gestione di una parte delle informazioni all'esterno mantenendone

altre all'interno.

La scelta del cloud ibrido è quella più diffusa e nel prossimo futuro riguarderà il 61% degli utenti. Sono quindi necessarie soluzioni sempre aggiornate e funzioni sempre più avanzate, in un contesto dove la **protezione delle informazioni riservate** ha un altissimo valore.

Le aziende si stanno confrontando con il problema della sua **integrazione con i software on-premise esistenti** e anche in questo caso la risposta è un cloud ibrido. Le soluzioni di cloud ibrido offrono in particolare **risorse di controllo e gestione** (policy, quote, limiti, regole di monitoraggio e di performance per il cloud pubblico) , **sicurezza** (controllo dell'accesso degli utenti, tramite sincronizzazione automatica delle directory utente delle applicazioni cloud e **on-premise**), **integrazione delle applicazioni** (approccio semplificato, "configurazione, non codifica") e **provisioning dinamico** (monitoraggio, provisioning e integrazione consentono di supportare il trasferimento dinamico dei carichi di lavoro da ambienti privati a cloud pubblici).

Il modello di cloud ibrido consente di utilizzare in modo rapido e a costi competitivi soluzioni disponibili come Software as a Service a condizione di poterle integrare in modo efficace con l'IT (cloud o tradizionale) on premise.

Conclusioni

Nel primo articolo di questa serie, ci siamo concentrati su una panoramica completa di tutti i concetti inerenti il mondo cloud. Oltretutto è stato importante individuare quali sono i vantaggi e gli svantaggi nell'adottare i vari modelli di servizio cloud (SaaS, PaaS e IaaS) e quali sono gli accorgimenti che occorre prendere nel momento in cui si decide di effettuare una "migrazione" verso il mondo cloud. Nella seconda parte di questa serie ci dedicheremo alla presentazione dei principali player di mercato attualmente presenti nell'ambito cloud, distinguendo tra quelli open e non, e facendo degli esempi di adozione di uno dei modelli di servizio.

Riferimenti

[1] Antonio Angelino, "Cloud Computing: analisi delle definizioni e descrizione delle tipologie di servizio presenti"

<http://ow.ly/dzJZX>

[2] Frederick Chong - Gianpaolo Carraro - Roger Wolter, "Architettura dati multi-tenant"

<http://msdn.microsoft.com/it-it/library/aa479086.aspx>

[3] Frederick Chong - Gianpaolo Carraro, "Architecture Strategies for Catching the Long Tail"

<http://msdn.microsoft.com/en-us/architecture/aa479069.aspx>

[4] La voce "Cloud Computing" su Wikipedia

http://it.wikipedia.org/wiki/Cloud_computing

[5] Tecnologie emergenti 2011 by Gartner

<http://blog.evoluzionetelematica.it/tecnologie-emergenti-2011-by-gartner/>

[6] Jeremy Geelan, "Twenty-One Experts Define Cloud Computing", Virtualization Journal

<http://virtualization.sys-con.com/node/612375?page=0,2>

[7] Antonio Angelino, "Cloud Computing: vantaggi e svantaggi per i clienti"

<http://ow.ly/dzKzc>

[8] Alberto Scotto, "Platform as a Service o PaaS"

<http://ow.ly/dzKGd>

[9] La voce "OVF" su Wikipedia

http://it.wikipedia.org/wiki/Open_Virtualization_Format

[10] Alessandro Diani, "Aspetti legali del Cloud Computing"

<http://www.quagliarella.com/rei20.html>

[11] Francesco Saliola "La testa fra le nuvole - Il parte: La riservatezza dei dati", MokaByte 174, giugno 2012

<http://ow.ly/dzKOM>

[12] Cloud Bursting

<http://www.clever-consulting.it/cms/News/cloud-bursting.html>

0 commenti

Ordina per **Principali**



Aggiungi un commento...

 Facebook Comments Plugin



Vittoria Caranna

Vittoria Caranna è nata a Rimini nel 1982. Lauretasi in Ingegneria Informatica presso l'Università degli studi di Bologna nel dicembre del 2007, da gennaio 2008 lavora per il Gruppo Imola. Svolge attività di consulenza, in particolare per quanto riguarda le tematiche architetture e di processo.

- **e-mail**

NELLO STESSO NUMERO

- **JavaEEFeatures**
- **javaportalPA**
- **JBossSeam-1**
- **componentdev-1**

- [soluzioncloud-1](#)
- [uxdesign-1](#)

NELLA STESSA SERIE

- [soluzioncloud-1](#)
- [soluzioncloud-3](#)
- [soluzioncloud-3](#)

UTILS

- [Stampa questa pagina](#)
- [Segnala errori](#)
- [Permalink](#)

ARCHIVIO

- [Luglio/Agosto 2012](#)
- [Giugno 2012](#)
- [Maggio 2012](#)
- [Aprile 2012](#)
- [Marzo 2012](#)
- [Febbraio 2012](#)
- [Gennaio 2012](#)
- [Dicembre 2011](#)
- [Novembre 2011](#)
- [Ottobre 2011](#)
- [Settembre 2011](#)
- [Luglio/Agosto 2011](#)
- [Giugno 2011](#)
- [Maggio 2011](#)
- [Aprile 2011](#)
- [Marzo 2011](#)
- [Febbraio 2011](#)
- [Gennaio 2011](#)
- [Dicembre 2010](#)
- [Novembre 2010](#)
- [Ottobre 2010](#)
- [Settembre 2010](#)
- [LuglioAgosto 2010](#)
- [Giugno 2010](#)

[Chi siamo](#)

[Contattaci](#)

MokaByte è un marchio registrato da MokaByte s.r.l. - Partita IVA:05039150486, CF:05039150486