

ITALIAN PATENT OFFICE

Document No.

102011901972045A1

Publication Date

20130210

Applicant

UNIVERSITA' DELLA CALABRIA

Title

SISTEMA PER IL RISPARMIO DI ENERGIA NEI DATA CENTER AZIENDALI.

Sistema per il
risparmio di energia nei data center aziendali

La presente invenzione riguarda un sistema per il risparmio di energia nei data center aziendali.

Più precisamente, la presente invenzione riguarda un metodo di assegnazione dei carichi computazionali ai vari server di un data center attraverso delle macchine virtuali ("Virtual Machine", VM), orientata alla riduzione dell'energia consumata nei data center, mantenendo allo stesso tempo un adeguato livello di servizio per i clienti del data center stesso.

L'avvento del Cloud Computing sta rappresentando un cambio di prospettiva radicale nella gestione delle infrastrutture informatiche per le imprese di qualsiasi dimensione e per le comunità scientifiche [Buy09]. Il vantaggio principale per le aziende è che non è più necessario gestire un proprio *data center*, con relativi costi e carichi amministrativi, ma è possibile affidare dati e computazioni alla "nuvola", cioè ai servizi offerti a pagamento dai data center di società specializzate. Ad esempio, è possibile eseguire un Web server su una Cloud invece che su computer locali, scelta che comporta notevoli vantaggi in termini di migliori garanzie di sicurezza e di disponibilità dei dati (ad esempio la difesa dagli hacker è gestita da professionisti), e rischi molto più bassi, se non nulli, di sottostima o sovrastima delle risorse necessarie.

E' necessario tuttavia porre molta attenzione al

problema del risparmio di energia nei data center, data l'enorme mole di energia necessaria per alimentare sia i server stessi che i sistemi di raffreddamento. Si stima che l'energia consumata per le infrastrutture di IT sia uguale, nei soli Stati Uniti, a 120 miliardi di kWh, corrispondenti a circa il 3% di tutta l'energia elettrica prodotta [Bell10]. Il consumo di energia ovviamente impatta sia sui costi delle società proprietarie dei data center che dei loro clienti, ed ha anche una forte ricaduta ambientale per le emissioni di CO₂.

Il consumo di energia è influenzato da due importanti fonti di inefficienza: da un lato i server sono in media utilizzati solo per il 20-30% della loro capacità, determinando un notevole spreco di risorse [Bar07]. Dall'altro lato un server con poco carico consuma circa il 70% della potenza consumata dallo stesso server a pieno carico [Gre09].

La tecnica maggiormente usata per alleviare il problema è il "consolidamento" delle applicazioni, cioè il loro raggruppamento sul minor numero possibile di server, in modo da poter disattivare o mettere in modalità a basso consumo i server non utilizzati [Bell11]. L'operazione di consolidamento è possibile grazie alla "virtualizzazione": le applicazioni sono associate a Macchine Virtuali (VM), che possono essere eseguite in contemporanea sullo stesso server, se questo ha le risorse necessarie, e possono essere "migrate" da un server all'altro a run-time [Ver08, Bell10].

Il problema della distribuzione ottimale delle VM ai server è riconducibile al problema del "bin packing", cioè dell'allocazione di un numero di oggetti di dimensione variabile (nel nostro caso VM con richieste di carico eterogenee) ad un numero minimo di contenitori (nel nostro caso i server) [Ver08, Sti09]. Purtroppo questo è un problema NP-hard [Yue91], cioè la cui soluzione ottima richiede un tempo esponenziale, ed è quindi impossibile da raggiungere in data center con decine o centinaia di server. Sono stati ideati algoritmi che sono in grado di pervenire a soluzioni sub-ottime [Maz10, Ver08, Bell10], ma che presentano alcuni importanti svantaggi:

- 1) si tratta di soluzioni centralizzate, cioè che richiedono che un manager centrale abbia informazioni sempre aggiornate sui singoli server, la qual cosa è ovviamente problematica in data center grandi e dinamici come quelli utilizzati per il Cloud Computing;
- 2) le soluzioni richiedono in genere la riallocazione contemporanea di un notevole numero di VM, cosa che può portare ad un decadimento delle prestazioni e dei livelli di servizio offerto agli utenti;
- 3) in ogni caso le tecniche euristiche utilizzate possono solo garantire di tenere in attività un numero di server che è superiore di circa il 20% rispetto al numero di server minimo [Yue91].

Il problema che si intende risolvere è quello della riduzione dell'energia consumata nei data center,

mantenendo allo stesso tempo un adeguato livello di servizio per i clienti del data center stesso. Nel caso di applicazioni di Cloud Computing, infatti, è necessario per rispettare i vincoli contrattuali stipulati con i clienti.

Il tema del consumo di energia nei data center sarà dominante per i prossimi anni, sia in termini di efficienza sia di controllo e gestione. Le strategie di risparmio energetico assumeranno diverse forme, dall'introduzione di tecnologie green - come i sistemi water-cooling o altri refrigeranti a livello delle macchine - alla gestione real-time dell'infrastruttura, che permette di spostare le risorse in base ai carichi di lavoro.

Anche lo spazio fisico occupato dai server ha la sua influenza. Al momento le aziende tendono a concentrare i server rack in uno spazio ridotto, ma questo fa salire la temperatura e di conseguenza la bolletta elettrica per il raffreddamento: secondo Gartner, la tendenza ad affollare i rack proseguirà fino a tutto il 2012 e il top management realizzerà presto che il costo energetico dell'IT, già alto oggi, non è che una frazione di quello che peserà sui bilanci domani. Gartner calcola che, fra tre anni, le spese in energia per un server saranno superiori al costo del server stesso. E agli attuali trend di crescita è probabile che i costi operativi dei server diventeranno presto uguali a quelli di capitale. Un rack da 40kW potrebbe costare fino a 5.400 dollari per server all'anno.

Di qui la necessità dei nuovi data center, con progettazioni all'avanguardia, per ridurre le spese sia di capitale sia operative e per ottenere maggiori prestazioni di calcolo per kilowatt. Alcune soluzioni per ridurre il consumo di energia nei data center possono essere l'utilizzo di raffreddamento a livello rack e row per le attrezzature a più alta densità, che può ridurre il consumo fino al 15%, mentre dimensionare il data center in base alle proprie esigenze di calcolo, ed espanderlo solo quando necessario, può abbattere le spese operative nel lungo termine del 10-30%.

Scopo della presente invenzione è quello di fornire un metodo per l'assegnazione dei carichi computazionali fra i server di un data center, che risolva i problemi e superi gli inconvenienti della tecnica anteriore.

E' ulteriore scopo specifico della presente invenzione quella di fornire gli apparati ed i mezzi necessari alla realizzazione del metodo scopo dell'invenzione

E' **oggetto** della presente invenzione un metodo per l'assegnazione di macchine virtuali ospitanti applicazioni ai server di una pluralità di server in un data center o centro di calcolo, per il consolidamento di dette macchine virtuali, ad ogni istante di tempo detta pluralità di server essendo suddivisa in un sottoinsieme di server attivi ciascuno ospitanti un certo numero di macchine virtuali ed un sottoinsieme di server inattivi, il metodo essendo caratterizzato dal

fatto di utilizzare un manager centrale del data center e una pluralità di manager locali di server in numero corrispondente a quello dei server di detta pluralità di server, e dal fatto che, vengono eseguite le seguenti fasi:

S1. almeno un manager locale relativo a detto sottoinsieme di server attivi decide singolarmente, effettuando un test di Bernoulli sulla base di una funzione di probabilità che varia al variare della percentuale di occupazione delle sue proprie risorse di calcolo ed è nulla per valori di occupazione delle risorse di calcolo più alti di una determinata soglia, se dichiarare la disponibilità ad accettare o no l'assegnazione di una nuova macchina virtuale, ed invia un relativo messaggio a detto manager centrale, il quale costruisce una lista di server disponibili;

e dal fatto che, quando è presente una nuova macchina virtuale (VM) da assegnare ad un server, detto manager centrale:

S2. se detta lista di server disponibili contiene almeno un elemento, sceglie a caso, tra gli elementi di detta lista, il server cui assegnare detta macchina virtuale da assegnare, e la assegna ad esso;

S3. se detta lista di server disponibili non contiene alcun elemento, attiva un server in detto sottoinsieme di server inattivi e gli assegna detta macchina virtuale da assegnare.

Preferibilmente secondo l'invenzione, la fase S1 è eseguita successivamente alla seguente fase preliminare:

S0. detto manager centrale invia in broadcast una richiesta di disponibilità ad ospitare detta macchina virtuale da assegnare al numero di manager locali relativo ad una porzione di detto sottoinsieme di server attivi, detto almeno un manager locale della fase S1 corrispondendo a detto numero di manager locali.

Preferibilmente secondo l'invenzione, detta richiesta di disponibilità è inviata a tutti i server di detto sottoinsieme di server attivi.

Preferibilmente secondo l'invenzione, la fase S1 è eseguita in maniera continua spontaneamente da tutti i server di detto sottoinsieme di server attivi.

Preferibilmente secondo l'invenzione, se detta macchina virtuale da assegnare deve essere trasferita da un server di detto sottoinsieme di server attivi, detta soglia predeterminata è più bassa per un predeterminato valore reale positivo Δ rispetto a quella del caso in cui detta macchina virtuale da assegnare non deve essere trasferita da un server di detto sottoinsieme di server attivi.

Preferibilmente secondo l'invenzione, detta funzione di probabilità è definita in modo che, se il server del relativo manager locale ha un carico computazionale più basso di una prima soglia o più alto di una seconda soglia, tende a rifiutare una nuova macchina virtuale da assegnare, mentre se ha un carico

computazionale intermedio tende ad accettarla, al fine di favorire detto consolidamento.

E' ulteriore oggetto specifico della presente invenzione un programma per elaboratore programma per computer, caratterizzato dal fatto di comprendere mezzi a codice atti ad eseguire, quando vengono fatti girare su una unità elettronica di elaborazione, le azioni del manager centrale come sopra descritte.

E' ulteriore oggetto specifico della presente invenzione un programma per computer, caratterizzato dal fatto di comprendere mezzi a codice atti ad eseguire, quando vengono fatti girare su una unità elettronica di elaborazione, le azioni di un manager locale come sopra descritte.

E' ulteriore oggetto specifico della presente invenzione un centro di calcolo o di archiviazione dati, comprendente una pluralità di server ed una unità elettronica centrale di gestione del centro di calcolo, caratterizzato dal fatto che su ogni server di detta pluralità di server è fatto girare il relativo programma per elaboratore oggetto dell'invenzione e su detta unità elettronica centrale di gestione è fatto girare il relativo programma per elaboratore oggetto dell'invenzione, essendo ulteriormente provvisti mezzi per la comunicazione ed il controllo tra detta unità elettronica centrale e i server di detta pluralità di server.

L'invenzione verrà ora descritta a titolo illustrativo ma non limitativo, con particolare riferimento ai disegni delle figure allegate, in cui:

- la figura 1 mostra l'assegnazione e la migrazione delle VM in un data center, secondo uno scenario tipico in cui è applicato il metodo dell'invenzione;
- la figura 2 mostra una connessione per l'accensione/spegnimento di un server da remoto, secondo la tecnica nota denominata "Wake On LAN" (WON);
- la figura 3 mostra esempi di funzione di probabilità di assegnamento, corrispondenti a differenti numeri di server coinvolti, secondo la presente invenzione.

La presente invenzione persegue l'obiettivo del consolidamento tramite due tipi di procedure statistiche, per l'assegnamento iniziale delle Macchine Virtuali (VM) ai server, e per la loro migrazione a run time.

Nello specifico, una nuova VM è assegnata ad uno dei server del data center a seguito dell'esecuzione di prove di Bernoulli, cioè test statistici che possono avere come risultato "vero" o "falso". Tali prove sono eseguite localmente sui singoli server, e la probabilità di successo dipende dalla utilizzazione corrente dei server, secondo delle apposite funzioni di probabilità. I server che hanno poco carico tendono a rifiutare nuove VM, così da scaricarsi del tutto ed essere poi disattivati. Analogamente, i server con molto carico tendono a rifiutare nuove VM, questa volta per evitare situazioni di sovraccarico che possono portare a decadimenti delle prestazioni ed alla

violazione degli accordi contrattuali stipulati con i clienti. Al contrario i server con carico intermedio tendono ad accettare nuove VM in modo da favorire il consolidamento.

L'invenzione inoltre prevede la possibilità di effettuare migrazioni a run time, cioè di trasferire le VM da un server all'altro quando questo può servire ad aumentare il consolidamento o a migliorare le prestazioni. Anche le migrazioni avvengono a seguito di prove di Bernoulli, ed anche in questo caso le decisioni, cioè se chiedere la migrazione di una VM e se accettare una VM migrata, sono effettuate in base ad informazioni gestite localmente sui vari server.

La figura 1 descrive lo scenario di riferimento per la procedura di assegnamento delle VM ai server e per la procedura di migrazione dinamica delle VM.

La richiesta di eseguire un'applicazione λ è trasmessa dal client al manager (o *front-end*) ξ del data center, che seleziona una VM adatta per quel tipo di applicazione, sulla base delle caratteristiche, ad esempio l'ammontare di risorse richieste (CPU, memoria, spazio di memorizzazione) ed il tipo di sistema operativo specificato dal client. In seguito, la VM è assegnata ad uno dei server mediante una procedura di assegnazione. Tale procedura, a grandi linee, è definita come segue.

Il front-end invia la richiesta in broadcast (si noti che le richieste in broadcast sono semplici e poco costose per questo tipo di ambienti) a tutti i server o ad un sottoinsieme di essi. Ogni server attivo esegue

il test di Bernoulli per decidere, sulla base esclusivamente di informazioni locali (carico computazionale), se accettare o no la richiesta. La probabilità di accettazione dipende dall'utilizzo corrente della CPU e dal massimo valore ammissibile per l'utilizzo della CPU (ad esempio, il 90%). La funzione di probabilità è definita in modo che i server con carico basso o alto tendano a rifiutare la nuova VM, mentre i server con carico intermedio tendano ad accettarla, al fine di favorire il consolidamento.

In maggiore dettaglio, e facendo riferimento alla figura 3, la funzione di probabilità del test di Bernoulli è definita per valori di utilizzo della CPU compresi tra 0 ed il massimo valore ammissibile (supponiamo il 90%), ed assume valori da 0 a 1. Per ottenere il risultato descritto, la funzione assume valori vicini a 0 (bassa probabilità di assegnamento) per valori di utilizzo della CPU vicini agli estremi, mentre assume valori vicini ad 1 (alta probabilità di assegnamento) per valori intermedi di utilizzo della CPU.

Dopo aver ricevuto la risposta dei vari server μ, v , il front-end assegna la VM a caso ad uno di essi, senza quindi dover eseguire nessun algoritmo complesso e centralizzato. Se nessuno dei server si dichiara disponibile questo significa, con la massima probabilità, che i server attualmente attivi non sono in grado di supportare il carico, e quindi viene attivato un nuovo server a cui sarà assegnata la VM.

Per quanto riguarda la procedura di migrazione, si

deve tener conto che il carico delle VM è fortemente variabile, si pensi al caso di una VM che ospita un server Web di un cliente. Quindi, anche a seguito di un assegnamento ottimale, le variazioni di carico possono richiedere la migrazione on line delle VM da un server all'altro.

La procedura di migrazione è preferibilmente iniziata localmente dal singolo server, nel momento in cui si accorge che il proprio carico è troppo basso (e quindi sarebbe meglio scaricare del tutto il server per poterlo disattivare) o troppo alto. Quando un server decide di far migrare una VM, esso effettua una procedura statistica simile a quella per l'assegnamento, solo che cambia leggermente la funzione di probabilità dei test di Bernoulli. Più dettagliatamente, la variazione riguarda la definizione del massimo valore ammissibile per l'utilizzo della CPU, valore che è leggermente più basso (ad esempio del 10%) che nella procedura di assegnamento. Questo serve a garantire che, nel caso di migrazione dovuta a sovraccarico, la VM migri verso un server con carico effettivamente minore, così da evitare una nuova immediata migrazione da sovraccarico.

Il risparmio di energia si basa sulla possibilità di spegnere ed accendere i server del data center in base al carico di richieste, tenendo in considerazione il duplice obiettivo di minimizzare il numero di server accesi e massimizzare la qualità del servizio ("Quality of Service", QoS).

Facendo riferimento alla figura 2, lo spegnimento

di un server può essere deciso dal manager locale che gira sul singolo server in base al carico, ed avviene mediante un comando di *shutdown*. Mentre lo spegnimento di un server può essere eseguito in locale via software, l'accensione da remoto richiede il supporto dell'hardware. Tale supporto è fornito dalle schede di rete Wake on LAN (WOL), basate su uno standard Ethernet che consente di avviare un computer in standby da una postazione remota, a patto di conoscerne l'indirizzo MAC, inviando alla LAN il cosiddetto *Magic Packet* (pacchetto magico). La scheda madre del computer (MainBoard o M/B) in questione dev'essere dotata di un apposito collegamento (detto WAKEUP-LINK) ed essere collegata alla scheda di rete attraverso uno speciale cavo. Da diversi anni però, se sia la scheda madre sia la scheda di rete supportano lo standard PCI 2.2, in genere il cavetto non è necessario dato che anche a computer spento gli slot pci vengono alimentati con una tensione di standby. Con l'introduzione della revisione 2.2 del PCI infatti viene aggiunto il supporto ai PME (Power Management Events) in modo che le schede PCI possano scambiare segnali anche a computer spento.

Dal punto di vista software, l'invenzione comprende due componenti fondamentali, uno per il manager del sistema ed uno per i singoli server. Il software è sviluppato in più linguaggi di programmazione (C, Java) e in diverse versioni per adattarsi ai diversi tipi di data center (Eucalyptus, VMWare, Amazon ecc.).

Il software del manager centrale svolge le

seguenti funzionalità:

1. inviare in broadcast la richiesta di accettare una VM;
2. scegliere a caso, tra i server disponibili, quello che dovrà allocare la VM;
3. in caso nessun server sia disponibile, attivare un nuovo server e chiedergli di allocare la VM.

Il software dei singoli server si occupa di monitorare il carico locale, e di fornire le primitive essenziali alla piattaforma (manager centrale), cioè:

1. attivare il server locale in seguito alla richiesta del manager;
2. spegnere il server locale quando non ci sono VM attive;
3. rispondere alle richieste in broadcast e dichiarare la propria disponibilità ad accettare una VM, in base al carico locale;
4. allocare una VM su richiesta del manager;
5. chiedere la migrazione di una VM (o, a seconda delle versioni, delegare questa attività al manager).

Il sistema può integrarsi con le piattaforme Cloud esistenti, ad esempio con "Elastic Compute Cloud" (EC2) di Amazon, un servizio di tipo IaaS ("Infrastructure-as-a-Service") basato sulla tecnologia di virtualizzazione XEN. Attraverso l'integrazione del manager con gli strumenti di gestione delle VM di EC2 (EC2ool), è possibile implementare le funzionalità necessarie del manager.

Esempio di procedura di assegnamento

Un client chiede l'esecuzione di un'applicazione al front-end, che associa l'applicazione ad una VM. Il front-end (manager centrale) invia la richiesta in broadcast ai server, chiedendo ad essi se sono disponibili ad eseguire la VM. Ogni server valuta l'utilizzo locale della CPU, ed effettua un test di Bernoulli per decidere se rendersi disponibile o meno.

Ad esempio, un server con basso utilizzo della CPU (supponiamo il 10%), valuta la funzione probabilistica di assegnamento in figura 3 che, visto il basso carico, assumerà un valore basso, ad esempio 0,03.

A questo punto occorre eseguire il test di Bernoulli. Si estrae quindi un numero casuale tra 0 ed 1 (o equivalentemente tra 0 e 100, trattandosi di probabilità); se questo valore estratto è minore di 0,03, allora il server si dichiarerà disponibile.

Questo vuol dire che, a seguito del test di Bernoulli, solo col 3% di probabilità il server si renderà disponibile ad eseguire la VM. Invece un server con carico intermedio (supponiamo il 60%), calcolerà un valore molto più alto della funzione di assegnamento: ad esempio il valore potrebbe essere 0.95, il che vuol dire che il server si renderà disponibile al 95%. Dopo aver ricevuto le dichiarazioni di disponibilità, il front-end sceglierà un server a caso tra i server disponibili, e assegnerà ad esso la VM.

Esempio di procedura di migrazione

La procedura di migrazione può essere iniziata in

maniera deterministica (ad esempio, se il server si accorge che il proprio carico è inferiore o superiore ad una soglia) o a seguito di un test probabilistico. La migrazione può poi essere effettuata in due modi: può essere il server stesso a mandare la richiesta in broadcast agli altri server, oppure il server può affidare la procedura al front-end, analogamente a quanto accade per la procedura di assegnamento. I server decidono se accettare la VM "migrante" ancora a seguito di un test di Bernoulli, analogamente a quanto visto prima, e l'ente che ha inviato l'invito (il front-end o il server di partenza) sceglie a caso tra i server che si sono resi disponibili. La funzione di probabilità è simile a quella usata per l'assegnamento, ma nel caso di migrazione dovuta a sovraccarico è modificata in modo tale che la VM dovrà migrare verso un server con utilizzo della CPU minore del server di partenza.

Una possibile variante per le due procedure è che non si invia l'invito in broadcast per ogni VM da assegnare/migrare, ma sono i singoli server che periodicamente, o in seguito ad eventi specifici locali (per esempio una variazione di carico significativa), comunicano spontaneamente al front-end la loro disponibilità ad accettare una VM, sempre a seguito di un test di Bernoulli. Quindi i server non aspettano l'invito, ma si propongono spontaneamente, così che il front-end può effettuare l'assegnamento, o completare la migrazione, già sapendo quali server sono

disponibili.

Vantaggi dell'invenzione

Tutti gli approcci per il consolidamento di macchine virtuali, a nostra conoscenza, si basano su approcci che presentano alcuni importanti problemi. Innanzitutto sono approcci basati su algoritmi centralizzati, che cercano, sulla base delle informazioni raccolte sull'intero data center, di distribuire in modo ottimale le VM sui vari server. Il problema è che l'assegnazione ottima è un tipico problema di "bin packing", che ha complessità NP-hard, cioè tempi di esecuzione esponenziali rispetto al numero di server e di VM. Esistono naturalmente degli algoritmi euristici, ad esempio basati sui metodi di programmazione intera, o sull'ordinamento delle VM secondo le loro richieste di carico, ma questi algoritmi possono avvicinarsi alla soluzione ottima solo entro un margine di circa il 20%. Inoltre, cosa ancora più importante, queste soluzioni richiedono che il manager centrale sia costantemente a corrente del carico di tutti i server e di tutte le applicazioni, cosa che è molto problematica in ambienti dinamici con decine, centinaia, o addirittura migliaia di server. Si tratta certamente di un approccio non ben scalabile.

Un'altra importante difficoltà è che gli algoritmi descritti in letteratura richiedono il riassegnamento periodico di molte VM, e quindi la loro migrazione simultanea. Questo può portare facilmente a un notevole

degrado delle prestazioni, e quindi al rischio di non ottemperare agli obblighi contrattati con i clienti.

Rispetto a quanto detto la soluzione secondo presente invenzione presenta almeno tre importanti vantaggi:

1. Grazie all'approccio statistico, la soluzione secondo la presente invenzione è **scalabile**, anzi è addirittura "over-scalable", cioè più che scalabile, nel senso che è tanto più efficiente quanto più grande è il data center. In particolare, i nostri test mostrano che a partire dagli 80 server è possibile avvicinarsi alla soluzione ideale entro un margine di non più del 3%, e le cose vanno ancora meglio per data center più grandi.
2. La soluzione secondo la presente invenzione è decentralizzata nel senso che *l'intelligenza* (per esempio la decisione se accettare una nuova VM o se richiedere una migrazione) risiede nei singoli server e non nel manager centrale. In particolare quest'ultimo non ha necessità di raccogliere informazioni aggiornate sui vari server, ma deve solo inviare delle richieste in broadcast e poi scegliere casualmente tra i server che si sono resi disponibili ad accettare una VM.
3. La soluzione secondo la presente invenzione richiede la migrazione *continua, graduale ed asincrona* delle VM. Non è mai necessario migrare diverse VM contemporaneamente. Quest'approccio ha notevoli vantaggi per qual riguarda il decadimento

della prestazioni legato alle migrazioni stesse.

L'invenzione può essere utilizzata per il risparmio di energia sui data center di dimensioni medio-grandi, cioè aventi da qualche decina di server fino a centinaia o migliaia di server.

Il mercato di riferimento è enorme. Nel seguito sono riportati il numero di server dei data center di alcune grosse società di IT e che operano nel settore del Cloud Computing:

- Intel: 100.000 server (dati di febbraio 2010);
- SoftLayer: 81.000 server (dati di maggio 2011);
- OVH: 80.000 server (dati di novembre 2010);
- Akamai Technologies: 73.000 server (dati di novembre 2010);
- Rackspace: 70.473 server (dati di maggio 2011)
- 1&1 Internet: 70.000 server (dati di febbraio 2010);
- Facebook: 60.000 server (dati di ottobre 2009);
- LeaseWeb: 36.000 server (dati di febbraio 2011);
- SBC Communications: 29.193 server (stime di Netcraft);
- Verizon: 25.788 server (stime di Netcraft);
- Time Warner Cable: 24.817 server (stime di Netcraft);
- AT&T: 20.268 server (stime di Netcraft);
- Google: oltre 450.000 server (stime per difetto del 2011);
- Microsoft: oltre 500.000 server (stime per difetto del 2011);

- Amazon: 40.000 server (stime per difetto del 2011);
- eBay: oltre 50.000 (stime del 2011);
- Yahoo: oltre 50.000 (stime del 2011);
- HP: oltre 380.000 (stime del 2011);
- IBM: oltre 50.000 (stime del 2011).

In Italia si calcola che ci siano circa 3.000 data center. In questo spaccato rientrano casi estremi che vanno dal piccolo data center di un'amministrazione locale agli oltre 15 data center degli Internet provider.

Uno studio di Oracle dello scorso maggio (<http://www.cwi.it>) individua quattro fasi di questa trasformazione: silo, consolidamento, ottimizzazione (integrazione dei vari strati) e infine Cloud. La media dei data center italiani si trova tra il primo e il secondo stadio, tra silo e consolidamento. Questo significa che gran parte delle aziende italiane appare lontana dal poter sfruttare pienamente il valore delle proprie risorse IT.

Esempi di come i data center italiani appaiano lontani dalla massima efficienza sono i seguenti: più del 75% eroga in media meno del 50% della potenza totale disponibile; più del 30% delle strutture non ha procedure automatizzate di system management, il che comporta l'utilizzo di molte risorse e scarsa flessibilità.

Ma i dati forse più eclatanti riguardano la virtualizzazione e la sostenibilità ambientale. Quasi il 40% delle aziende ha ammesso di aver virtualizzato

meno del 10% dei server hardware presenti nel data center, e un altro 20% abbondante tra il 10 e il 29% dei server.

Per le aziende italiane emerge la necessità di un cambiamento. Date le problematiche generate da infrastrutture complesse e inefficienti, secondo Oracle il 61% delle imprese prevede la necessità di implementare una nuova struttura di data center entro i prossimi due anni.

Per nuova struttura si intende un data center gestito internamente o anche esternalizzato. Il **55%** delle aziende italiane possiede **un solo data center in-house**. Il **23%** ha alcuni data center in-house e nessuno esterno. Il **15%** ha un mix composto da un data center principale interno più alcuni esterni. Infine il **7%** possiede un mix di alcuni data center in-house più alcuni esterni.

Per quanto riguarda lo sviluppo del Cloud Computing, si deve inoltre tener conto che da parte di molte società c'è soprattutto interesse a servizi di "Private Cloud". Si tratta di ambienti specifici di IT in cui accesso e condivisione sono privati. Un Cloud privato può trovarsi sia presso l'azienda stessa, sia esternamente in hosting da un provider esterno. C'è anche il Cloud "ibrido", che combina i servizi IT esterni, cioè forniti da società specializzate, con altri gestiti direttamente dal cliente. Il comportamento tipico delle aziende è in genere improntato alla prudenza: si parte del Cloud privato e si passa solo successivamente ai sistemi ibridi o

esterni. Da ciò deriva che la problematica del risparmio di energia non è importante solo per le società specializzate, ma anche per la gestione dei data center delle società che si affidano a sistemi di Cloud Computing privati.

Per quanto riguarda l'Italia, si prevede che il volume di affari per i prossimi dodici mesi sarà di 287 milioni di euro. Una recente ricerca di Nextvalue (Cloud Computing Report, aprile 2011) stima che il 61% delle grandi imprese italiane (e l'80% in Europa) ha adottato o adotterà nei prossimi 12 mesi progetti di Cloud Computing. Per quanto riguarda le PMI, un rapporto della School Of Management del Politecnico di Milano, presentato a maggio, rileva che il livello di diffusione del Cloud nelle piccole e medie aziende è ancora nell'ordine del 2-3%, ma è previsto in grande crescita. Le aspettative delle aziende rispetto al Cloud sono molto alte e riguardano principalmente il contenimento dei costi, su hardware e personale e, in seconda battuta, la velocità di attivazione e la flessibilità.

Bibliografia

- [Bell11] Anton Beloglazov, Rajkumar Buyya, Young Choon Lee, Albert Y. Zomaya: A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems. *Advances in Computers* 82: 47-111 (2011)
- [Buy09] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic.

- Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 25(6):599-616, June 2009.
- [Bel10] Anton Beloglazov and Rajkumar Buyya. Energy efficient allocation of virtual machines in cloud data centers. In *10th IEEE/ACM Int. Symp. on Cluster Computing and the Grid, CCGrid 2010*, pages 577-578, 2010.
- [Bar07] Luiz André Barroso and Urs Holzle. The case for energy-proportional computing. *IEEE Computer*, 40(12):33-37, December 2007.
- [Gre09] Albert Greenberg, James Hamilton, David A. Maltz, and Parveen Patel. The cost of a cloud: research problems in data center networks. *SIGCOMM Comput. Commun. Rev.*, 39(1):68-73, 2009.
- [Ver08] Akshat Verma, Puneet Ahuja, and Anindya Neogi. pMapper: Power and migration cost aware application placement in virtualized systems. In Valrie Issarny and Richard E. Schantz, editors, *Middleware 2008, ACM/IFIP/USENIX 9th International Middleware Conference, Leuven, Belgium, December 1-5, 2008, Proceedings, volume 5346 of Lecture Notes in Computer Science*, pages 243-264. Springer, 2008.
- [Sti09] M. Stillwell, D. Schanzenbach, F. Vivien, H. Casanova, Resource allocation using virtual clusters, in: *Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing*

and the Grid (CCGrid 2009), Shanghai, China, 2009, pp. 260-267.

[Yue91] Minyi Yue. A simple proof of the inequality $FFD(L) \leq 11/9 OPT(L) + 1$, for all L for the FFD bin-packing algorithm. *Acta Mathematicae Applicatae Sinica*, 7(4):321-331, 1991.

[Maz10] Michele Mazzucco, Dmytro Dyachuk, and Ralph Deters. Maximizing cloud providers' revenues via energy aware allocation policies. In 10th IEEE/ACM Int. Symp. on Cluster Computing and the Grid, CCGrid 2010, pages 131-138, 2010.

In quel che precede sono state descritte le preferite forme di realizzazione e sono state suggerite delle varianti della presente invenzione, ma è da intendersi che gli esperti del ramo potranno apportare modificazioni e cambiamenti senza con ciò uscire dal relativo ambito di protezione, come definito dalle rivendicazioni allegate.

Barzanò & Zanardo Roma S.p.A.

RIVENDICAZIONI

1) Metodo per l'assegnazione di macchine virtuali (VM) ospitanti applicazioni (λ) ai server di una pluralità di server (μ, v) in un data center o centro di calcolo, per il consolidamento di dette macchine virtuali, ad ogni istante di tempo detta pluralità di server (μ, v) essendo suddivisa in un sottoinsieme di server attivi (μ) ciascuno ospitanti un certo numero di macchine virtuali (VM) ed un sottoinsieme di server inattivi (v), il metodo essendo caratterizzato dal fatto di utilizzare un manager centrale (ξ) del data center e una pluralità di manager locali di server in numero corrispondente a quello dei server di detta pluralità di server (μ, v), e dal fatto che, vengono eseguite le seguenti fasi:

S1. almeno un manager locale relativo a detto sottoinsieme di server attivi (μ) decide singolarmente, effettuando un test di Bernoulli sulla base di una funzione di probabilità che varia al variare della percentuale di occupazione delle sue proprie risorse di calcolo ed è nulla per valori di occupazione delle risorse di calcolo più alti di una determinata soglia, se dichiarare la disponibilità ad accettare o no l'assegnazione di una nuova macchina virtuale (VM), ed invia un relativo messaggio a detto manager centrale (ξ), il quale costruisce una lista di server disponibili;

e dal fatto che, quando è presente una nuova macchina

virtuale (VM) da assegnare ad un server, detto manager centrale (ξ):

S2. se detta lista di server disponibili contiene almeno un elemento, sceglie a caso, tra gli elementi di detta lista, il server cui assegnare detta macchina virtuale (VM) da assegnare, e la assegna ad esso;

S3. se detta lista di server disponibili non contiene alcun elemento, attiva un server in detto sottoinsieme di server inattivi (v) e gli assegna detta macchina virtuale (VM) da assegnare;

in cui se ad un dato istante il manager locale non fa girare alcuna macchina virtuale sul relativo server locale, quest'ultimo di disattiva e si spegne ed è quindi incluso nel sottoinsieme di server inattivi (v), così da minimizzare il numero di server attivi (μ).

2) Metodo secondo la rivendicazione 1, caratterizzato dal fatto che la fase S1 è eseguita successivamente alla seguente fase preliminare:

S0. detto manager centrale (ξ) invia in broadcast una richiesta di disponibilità ad ospitare detta macchina virtuale (VM) da assegnare al numero di manager locali relativo ad una porzione di detto sottoinsieme di server attivi (μ),

detto almeno un manager locale della fase S1 corrispondendo a detto numero di manager locali.

3) Metodo secondo la rivendicazione 2, caratterizzato dal fatto che detta richiesta di disponibilità è inviata a tutti i server di detto

sottoinsieme di server attivi (μ).

4) Metodo secondo la rivendicazione 1, caratterizzato dal fatto che la fase S1 è eseguita in maniera continua spontaneamente da tutti i server di detto sottoinsieme di server attivi (μ).

5) Metodo secondo una qualsiasi delle rivendicazioni da 1 a 4, caratterizzato dal fatto che se detta macchina virtuale (VM) da assegnare deve essere trasferita da un server di detto sottoinsieme di server attivi (μ), detta soglia predeterminata è più bassa per un predeterminato valore reale positivo Δ rispetto a quella del caso in cui detta macchina virtuale (VM) da assegnare non deve essere trasferita da un server di detto sottoinsieme di server attivi (μ).

6) Metodo secondo una qualsiasi delle rivendicazioni da 1 a 5, caratterizzato dal fatto che detta funzione di probabilità è definita in modo che, se il server del relativo manager locale ha un carico computazionale più basso di una prima soglia o più alto di una seconda soglia, tende a rifiutare una nuova macchina virtuale (VM) da assegnare, mentre se ha un carico computazionale intermedio tende ad accettarla, al fine di favorire detto consolidamento.

7) Programma per computer, caratterizzato dal fatto di comprendere mezzi a codice atti ad eseguire, quando vengono fatti girare su una unità elettronica di elaborazione, le azioni del manager centrale di una qualsiasi delle rivendicazioni da 1 a 6.

8) Programma per computer, caratterizzato dal fatto di comprendere mezzi a codice atti ad eseguire, quando vengono fatti girare su una unità elettronica di elaborazione, le azioni di un manager locale di una qualsiasi delle rivendicazioni da 1 a 6.

9) Centro di calcolo o di archiviazione dati, comprendente una pluralità di server (μ, ν) ed una unità elettronica centrale di gestione (ξ) del centro di calcolo, caratterizzato dal fatto che su ogni server di detta pluralità di server è fatto girare il programma per elaboratore secondo la rivendicazione 7 e su detta unità elettronica centrale di gestione è fatto girare il programma per elaboratore secondo la rivendicazione 8, essendo ulteriormente provvisti mezzi per la comunicazione ed il controllo tra detta unità elettronica centrale e i server di detta pluralità di server.

Barzanò & Zanardo Roma S.p.A.

CLAIMS

1) Method for the assignation of virtual machines (VM) hosting applications (λ) to servers of a plurality of servers (μ, ν) in a data or calculation centre, for the consolidation of said virtual machines, to each time instant said plurality of servers (μ, ν) being subdivided in a subset of active servers (μ), each hosting a certain number of virtual machines (VM) and a subset of inactive servers (ν), the method being characterised in that it utilises a central manager (ξ) of the data centre and a plurality of local managers in a number corresponding to that of the servers of said plurality of servers (μ, ν), and in that, the following steps are executed:

S1. At least a local manager relevant to said subset of active servers (μ) decides individually, by performing a Bernoulli test on the basis of a probability function that depends on the occupation percentage of its own calculation resources and is null for calculation resources occupation values higher than a pre-defined threshold, whether to declare its availability to accept or not the assignation of a new virtual machine (VM), and sends a relevant message to said central manager (ξ), which constructs a list of available servers;

and in that, when there is a new virtual machine (VM) to be assigned to a server, said central manager (ξ):

S2. if said list of available servers contain at least

an element, chooses randomly, among the elements of said list, the server to which said new virtual machine (VM) is to be assigned, and assigns it to the former;

S3. if said list of available servers does not contain any element, activates a server in said subset of inactive servers (v) and assigns to it said virtual machine (VM) to be assigned.

2) Method according to claim 1, characterised in that step S1 is performed after the following preliminary step:

S0. Said central manager (ξ) broadcasts a request of availability to host said new virtual machine (VM) to be assigned to the number of local managers of a portion of said subset of active servers (μ), Said at least a local manager of step S1 corresponding to said number of local managers.

3) Method according to claim 2, characterised in that said request of availability is sent to all the servers of said subset of active sets (μ).

4) Method according to claim 1, characterised in that step S1 is executed in a continuous manner spontaneously by all the servers of said subset of active servers (μ).

5) Method according to any claim 1 to 4, characterised in that if said virtual machine (VM) to be assigned is to be transferred from a server of said subset of active servers (μ), said pre-defined threshold is smaller, by a pre-determined real positive

value Δ , than that of the case when said virtual machine (VM) to be assigned is not to be transferred from a server of said subset of active servers (μ).

6) Method according to any claim 1 to 5, characterised in that said probability function is defined in such a way that, if a server relevant to a local manager has a computational load smaller than a first threshold or larger than a second threshold, it tends to refuse a new virtual machine (VM) to be assigned, whilst if it has an intermediate computational load it tends to accept it, to the end of favouring said consolidation.

7) Computer program, characterised in that it comprises code means that are suitable, when they are made running on a elaboration electronic unit, the actions of a central manager according to any claim 1 to 6.

8) Computer program, characterised in that it comprises code means that are suitable, when they are made running on a elaboration electronic unit, the actions of a local manager according to any claim 1 to 6.

9) Calculation or data centre, comprising a plurality of servers (μ, ν) and a management central electronic unit (ξ) of the calculation centre, characterised in that on each server of said plurality of servers the computer program according to claim 7 is made running and on said management central electronic unit the computer program according to claim 8 is made running, being further provided means for the

communication and control between said management central electronic unit and the servers of said plurality of servers.

Barzanò & Zanardo Roma S.p.A.

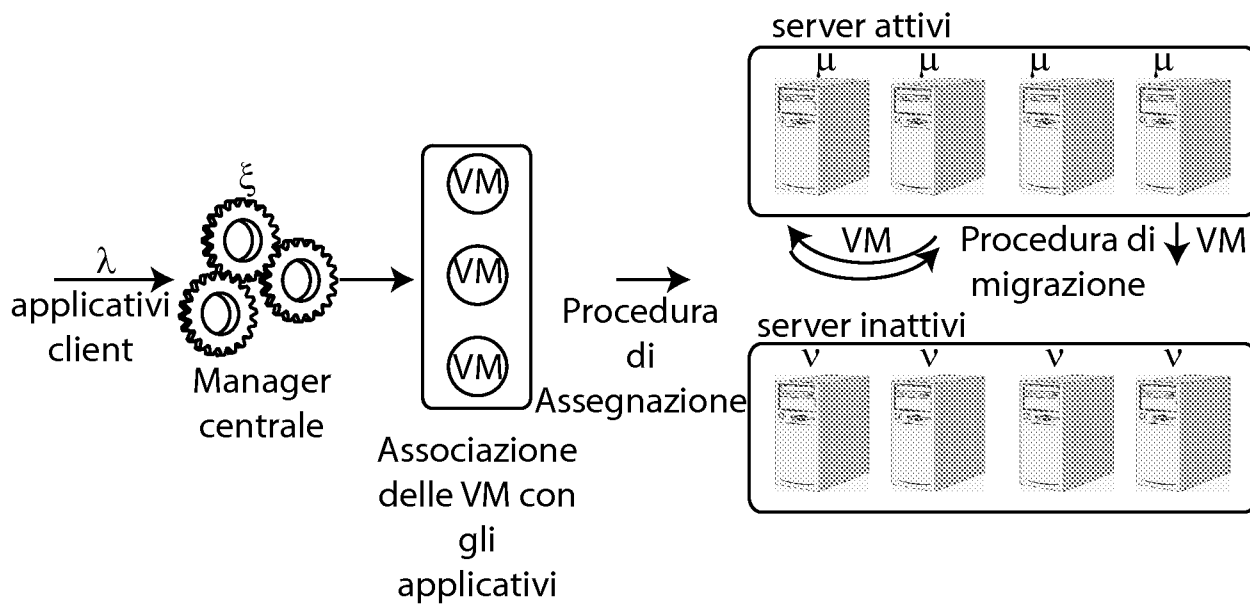


Fig. 1

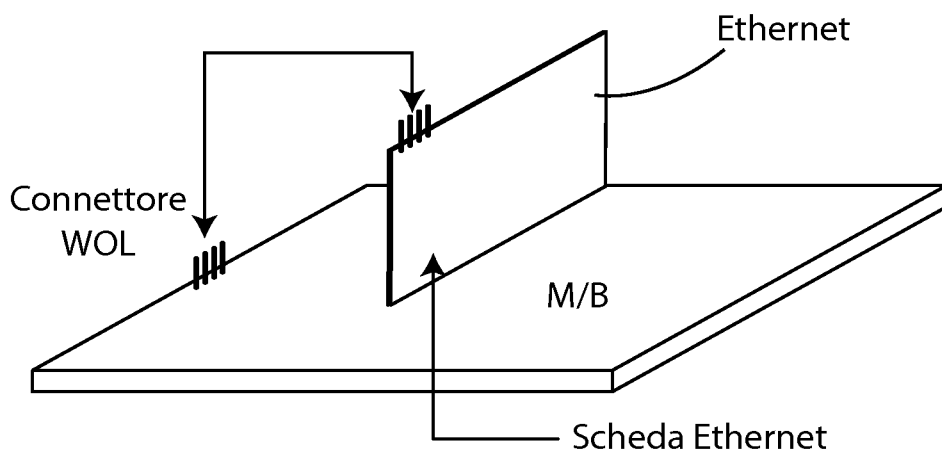


Fig. 2

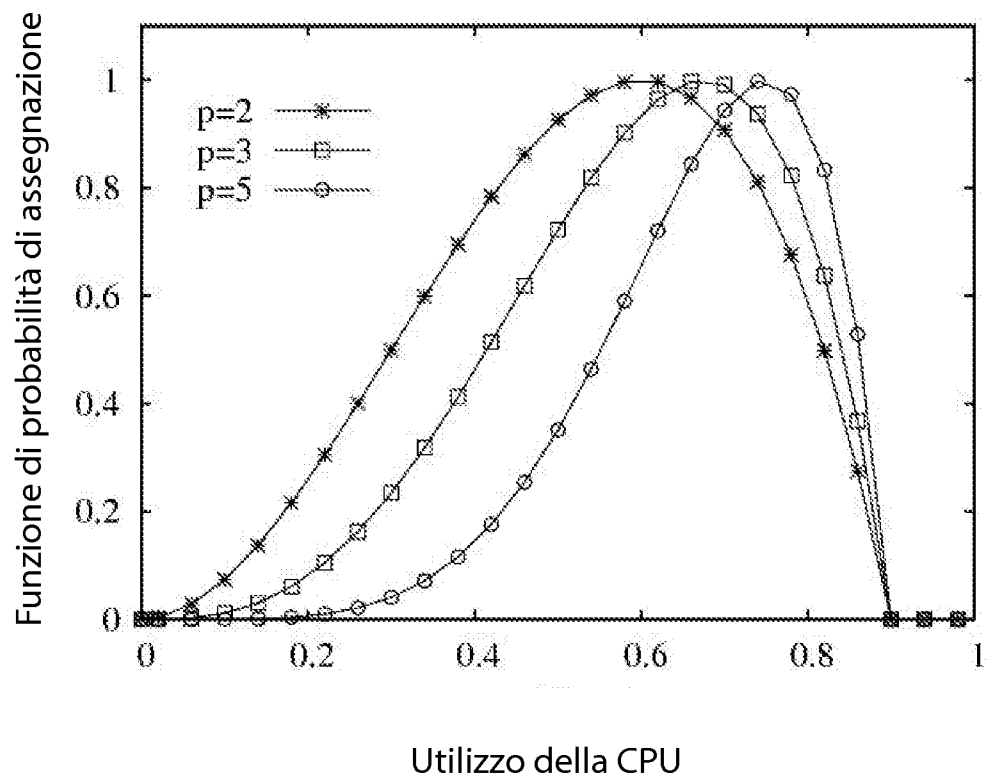


Fig. 3