

ALMA MATER STUDIORUM · UNIVERSITÀ DI
BOLOGNA

FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI
Corso di Laurea Triennale in Informatica

**Studio e valutazione
di procedure di consolidamento
per server fisici in strutture virtuali**

Tesi di Laurea in Progetto di Sistemi Virtuali

**Relatore:
Chiar.mo Prof.
Renzo Davoli**

**Presentata da:
Luca Serravalli**

**Sessione II
Anno Accademico 2010/2011**

*You can be anything you want to be
Just turn yourself into anything you think that you could ever be
Be free with your tempo, be free be free
Surrender your ego - be free, be free to yourself . . .*

Introduzione

Il continuo aumentare della richiesta di servizi informatici necessari per soddisfare le esigenze di business, ha portato con il passare degli anni ad un continuo e costante incremento del numero di server utilizzati nei data center delle organizzazioni. Gestire una struttura informatica composta da centinaia di macchine fisiche non è un compito facile, né dal punto di vista tecnico né dal punto di vista economico: personale tecnico numeroso, considerevole quantità di tempo sprecata per amministrare le macchine, costi elevati di manutenzione hardware, consumo elettrico proibitivo ed enormi spazi necessari per ospitare i server sono alcuni dei problemi che un amministratore di tali strutture si trova a dover gestire, senza contare che molto spesso i singoli server sono sottoutilizzati e sfruttano soltanto una minima parte delle risorse realmente disponibili. Quando questa crescita arriva al punto da rendere non efficiente ed antieconomica la gestione della struttura, occorre una soluzione per invertire la tendenza e rimediare alle problematiche sorte.

La soluzione è un processo di razionalizzazione e riorganizzazione delle risorse che prende il nome di *Server Consolidation* (consolidamento di server), tema inequivocabilmente legato al concetto di *Virtualizzazione*; la storia recente ha dimostrato quanto importante sia questo settore e quanti siano gli attori che vi partecipano: la comunità scientifica ed il mondo *Open Source*, singoli sviluppatori, aziende e multinazionali, grazie ai quali al giorno d'oggi esistono innumerevoli prodotti che hanno reso possibile e praticabile il consolidamento di macchine fisiche in strutture virtuali.

Scopo di questa tesi è dunque analizzare il fenomeno, valutando quali

siano le tecnologie e le procedure coinvolte, e cercando in particolare di fornire una visione organica e d'insieme del processo, di cosa comporti e di come lo si esegua. In primo luogo si darà quindi una descrizione di *cosa* sia il consolidamento di server e di quali aspetti coinvolga; si analizzerà quindi *come* esso venga realizzato attraverso la virtualizzazione, illustrandone le caratteristiche e le tecnologie al riguardo; infine, si fornirà un modello descrittivo suddiviso in fasi che consenta di avere una visione globale del processo, a partire dal passo iniziale fino al passo finale. Di seguito la struttura di questa tesi:

Capitolo 1: Introduzione al consolidamento di server

Nel primo capitolo si discuterà delle problematiche e dei costi derivanti dalla gestione di parchi macchine non consolidati, mostrando in generale in cosa consiste il consolidamento di server e che cosa comporta. Verranno messi in evidenza i principali vantaggi e svantaggi che si ottengono con tale processo, senza entrare per il momento nel dettaglio del *come* questo venga fatto, quindi si discuterà brevemente di alcuni casi reali in cui sono state applicate tecniche di consolidamento.

Capitolo 2: Tecniche e soluzioni per il consolidamento

Il secondo capitolo si concentrerà sul fornire un'idea generale, ma il più completa possibile, della tecnica principale che permette di ottenere un sistema consolidato: la virtualizzazione. Si valuteranno quindi le caratteristiche delle diverse forme di virtualizzazione - Bare-metal/Hosted Hypervisor, Trap-and-emulate/Full virtualization/Paravirtualization/Hardware-assisted virtualization - e si forniranno esempi di alcuni prodotti che realizzano le diverse forme. Infine, si farà un passo in avanti sul concetto di virtualizzazione e si parlerà di Cloud Computing, valutandone le caratteristiche principali e gli effetti sul tema del consolidamento.

Capitolo 3: Analisi delle fasi del processo

Nel capitolo conclusivo si cercherà di fornire una visione complessiva di quali siano i passi da compiere per consolidare una struttura informatica. Verrà fornito un modello articolato in cinque fasi denominate *Definizione degli obiettivi*, *Raccolta dati ed analisi dei workload*, *Progettazione della struttura*, *Migrazione verso la struttura virtuale* e *Gestione della struttura*; queste sono ognuna caratterizzata da obiettivi e operazioni da compiere ben precise, attraverso le quali verranno evidenziate problematiche, soluzioni e scelte da prendere in considerazione quando si intraprende un processo di consolidamento.

Indice

Introduzione	i
1 Introduzione al consolidamento di server	1
1.1 La proliferazione ed i suoi costi	2
1.2 Quali vantaggi si ottengono	3
1.2.1 Alta affidabilità e continuità di servizio	3
1.2.2 Scalabilità e facilità di gestione	4
1.2.3 Manutenzione e ciclo di vita semplificati	4
1.3 Svantaggi e possibili rischi	5
1.3.1 Proliferazione virtuale	5
1.3.2 Disaster recovery costoso per i server fisici	6
1.3.3 Problemi di licenze e di supporto	6
1.4 Alcuni casi reali di strutture consolidate	7
2 Tecniche e soluzioni per il consolidamento	11
2.1 Bare-metal hypervisor (Type 1)	13
2.2 Hosted hypervisor (Type 2)	13
2.3 Istruzioni privilegiate e trap and emulate	15
2.4 Full virtualization	16
2.5 Paravirtualization	17
2.5.1 Xen	18
2.6 Hardware-assisted Virtualization	19
2.6.1 KVM	20
2.7 Consolidamento e outsourcing: Cloud computing	21

2.8	Caratteristiche	22
2.9	Consolidare nella nuvola	23
2.9.1	Elasticità e provisioning rapido	23
2.9.2	Lock-in e riservatezza	24
3	Analisi delle fasi del processo	25
3.1	Definizione degli obiettivi	25
3.1.1	Approccio di contenimento	26
3.1.2	Approccio strutturale	27
3.2	Raccolta dati e analisi workload	27
3.2.1	Monitoring tools	28
3.3	Progettazione della struttura	29
3.3.1	Opzioni di consolidamento	29
3.3.2	Overhead della virtualizzazione	32
3.3.3	Best practices	33
3.4	Migrazione verso la struttura virtuale	34
3.4.1	Physical to Virtual Migration	35
3.4.2	Virtual to Virtual Migration	36
3.5	Gestione della struttura	37
	Conclusioni	39
	Bibliografia	41

Elenco delle figure

1.1	Analisi utilizzo server CeSIA	8
2.1	Struttura macchina virtuale	13
2.2	Bare-metal hypervisor	14
2.3	Hosted hypervisor	14
2.4	Livelli di privilegio reali	16
2.5	Full virtualization	17
2.6	Paravirtualization	18
2.7	Architettura basata su Xen	19
2.8	Hardware-assisted virtualization	20
2.9	Architettura basata su KVM	21
3.1	DMZ in a Box	30
3.2	Workload complementari	32
3.3	Live migration	38

Capitolo 1

Introduzione al consolidamento di server

Per consolidamento si intende in generale un processo di ottimizzazione e riorganizzazione delle risorse all'interno di una struttura informatica. Si parla di consolidamento in almeno tre casi diversi:

Consolidamento geografico quando cioè si raccolgono macchine dislocate su datacenter diversi in un'unica struttura centralizzata;

Consolidamento di applicazioni se si riduce il numero di applicazioni diverse utilizzate, sfruttando processi di standardizzazione;

Consolidamento di server quando si riduce il numero di macchine utilizzate mantenendo gli stessi servizi;

La tipologia di consolidamento che in particolare sarà oggetto di discussione riguarda quello effettuato sui server: quando cioè si adottano strategie volte a diminuire il numero di macchine fisiche utilizzate, in modo che servizi e applicazioni che prima risiedevano su server fisicamente separati confluiscono all'interno di una stessa macchina.

Nel corso degli ultimi decenni il tema della riduzione delle macchine fisiche è diventato sempre più d'uso comune: i progressi svolti nel settore della virtualizzazione - la tecnica principale, di cui si parlerà in seguito, che consente di

attuare consolidamento di server - hanno permesso di soddisfare le richieste di riduzione di costo e di semplificazione e ottimizzazione delle risorse e dei servizi di un'organizzazione. Attraverso il consolidamento è possibile contrastare quel problema comunemente chiamato *proliferazione di server*, ossia l'aggiunta incontrollata e non giustificata di nuove macchine all'infrastruttura già esistente.

1.1 La proliferazione ed i suoi costi

A partire dagli anni ottanta, la possibilità di acquistare server a prezzi sempre più competitivi, unita ad una sempre maggiore tendenza da parte delle organizzazioni a dotarsi di infrastrutture tecnologiche - in particolare da parte di aziende in corsa per soddisfare le crescenti richieste di mercato - ha portato alla pratica comune di acquistare nuove macchine ogni volta che si doveva fornire un nuovo servizio o implementare nuove applicazioni. Come risultato si ottiene che spesso una macchina lavora sfruttando in media soltanto una piccola parte delle proprie capacità, con percentuali che oscillano tra il 10% ed il 30%.

Con il passare del tempo, un approccio di questo tipo causa l'aumento incontrollato della spesa necessaria per gestire una struttura realizzata con quest'ottica, oltre a rappresentare uno spreco energetico ed un fattore di danno per l'ambiente. Si possono in particolare evidenziare i seguenti costi:

- Acquisto di nuovo hardware
- Manutenzione
- Raffreddamento
- Alimentazione

La spesa iniziale necessaria per l'acquisto di un nuovo server non è che la punta dell'iceberg del costo con cui questa nuova macchina inciderà sull'organizzazione. Un nuovo server significa prima di tutto una macchina in

più che l'amministratore di sistema deve gestire durante tutto il ciclo di vita; comporta anche una spesa maggiore in energia elettrica per l'alimentazione e per il raffreddamento, oltre alle spese logistiche necessarie per gestire armadi rack sempre più affollati e numerosi; al termine del ciclo di vita poi, sarà necessario personale che si occupi della migrazione dei dati verso un nuovo server. Infine, avere parchi macchine di ampie dimensioni implica spese maggiori per garantire situazioni di *High Availability* e di *Fault Tolerance*, oltre alla necessità di assumere personale sempre più numeroso.

È quindi evidente che ad ogni aggiunta di una nuova macchina corrisponde una serie di costi che potrebbero essere limitati ed in parte evitati sfruttando tecniche di consolidamento.

1.2 Quali vantaggi si ottengono

Il vantaggio più ovvio consiste nel risparmio economico ottenuto riducendo il numero di macchine, come evidenziato nel paragrafo precedente. Dal punto di vista delle risorse fisiche infatti, l'unica spesa aggiuntiva coincide con l'acquisto iniziale dei nuovi server - tipicamente più performanti e quindi più costosi - e del resto delle strutture eventualmente necessarie. L'utilizzo ottimizzato delle risorse fisiche non è però l'unico vantaggio che deriva da un processo di consolidamento: virtualizzare una serie di macchine e gestirle all'interno dello stesso server porta benefici anche dal punto di vista della manutenzione, della gestione dei guasti, della scalabilità e dell'amministrazione di sistema. Di seguito alcuni esempi.

1.2.1 Alta affidabilità e continuità di servizio

Gestire i disservizi di una macchina virtualizzata risulta molto più agevole della rispettiva controparte fisica: non è necessario acquistare hardware ridondante in quanto la replicazione avviene a livello software, così come non è necessario eseguire *downtime* pianificati per attività di manutenzione sui nuovi server virtuali.

Risulta inoltre più agevole anche il mantenimento di sistemi e software cosiddetti *legacy*, cioè quel tipo di software obsoleto e che tipicamente incorre più di frequente in guasti e malfunzionamenti. Ospitare tali applicativi in server dedicati, per evitare che possano arrecare danno alle prestazioni di altri componenti, così come essere costretti a mantenere hardware superato perché richiesto da sistemi legacy, sono alcune delle problematiche che possono essere in gran parte risolte senza grandi sforzi.

1.2.2 Scalabilità e facilità di gestione

Una struttura consolidata, nella quale le macchine virtuali possono essere spostate da un server all'altro, consente di adattare le risorse in modo dinamico al variare della richiesta di maggiore o minore capacità di calcolo; questo permette di gestire ad un livello di granularità più fine le risorse da destinare ai singoli servizi, sfruttando intensivamente le caratteristiche dei server fisici. Analogamente, poter gestire le risorse a livello logico in modo centralizzato consente di semplificare le operazioni quotidiane come la gestione dei *backup*.

1.2.3 Manutenzione e ciclo di vita semplificati

Come primo aspetto va sottolineato che una struttura virtuale consente di evitare la fase finale del ciclo di vita di una macchina: la sua dismissione. Questo consente, tra le varie cose, di evitare i costi per lo smaltimento dei componenti inquinanti per l'ambiente, pratica che per un'organizzazione che gestisce centinaia di calcolatori si traduce in un risparmio non indifferente; si consideri inoltre che in certi ambiti si hanno cicli di vita molto rapidi, nell'ordine dei 3-5 anni, alla cui scadenza molte macchine andrebbero sostituite interamente.

Allo stesso modo viene evitata la manutenzione ordinaria da compiere sull'hardware, così come viene semplificata la procedura di *provisioning*, cioè di messa in funzione di un sistema vergine e di installazione di tutte le ap-

plicazioni necessarie; non va poi dimenticato che si evita tutta la procedura di ordine, acquisto, consegna e montaggio delle nuove macchine.

1.3 Svantaggi e possibili rischi

Un processo di consolidamento mal pianificato può portare a seri rischi per l'organizzazione che intraprende il cammino, in primo luogo dal punto di vista economico: sottovalutare la potenza di calcolo necessaria per i nuovi server fisici, collocare un numero eccessivo di macchine virtuali sulla stessa macchina fisica, non prevedere picchi fisiologici di richieste di risorse e la mancanza di una strategia per far fronte a situazioni critiche, sono soltanto alcuni dei motivi che possono portare ad un aumento dei costi, in particolare qualora l'organizzazione si trovi a non poter erogare servizi che ha garantito.

Oltre a quei problemi imputabili alla gestione scorretta del processo, esiste comunque la possibilità di dover affrontare situazioni impreviste che possono provocare disagi.

1.3.1 Proliferazione virtuale

Una situazione da evitare è quella in cui, analogamente a quanto detto in precedenza sulla proliferazione fisica, si eccede nella creazione di nuove macchine virtuali. Se ad esempio i servizi vengono sviluppati in macchine virtuali in rapporto di 1:1, sicuramente si degradano gli effetti positivi del consolidamento in quanto le risorse necessarie ad una macchina virtuale e la sua manutenzione, per quanto minori esse siano rispetto alla controparte fisica, incidono comunque sul bilancio totale; allo stesso tempo si rischia di aumentare la complessità logica dell'infrastruttura, creando troppe ramificazioni e gerarchie di macchine virtuali che diventano difficili da mantenere e organizzare.

1.3.2 Disaster recovery costoso per i server fisici

La gestione di situazioni critiche che coinvolgono le macchine fisiche può risultare più costosa di quanto lo fosse in precedenza, anche a causa del semplice fatto che si tratta di hardware più costoso di quello che si utilizzava prima. In una struttura consolidata si può inoltre avere a che fare con un server fisico sul quale girano decine di macchine virtuali: se prima si verificava un guasto su di un server, rendendolo non più operativo, tipicamente si verificavano disservizi su una particolare applicazione o su quelle poche funzionalità ospitate dalla macchina caduta; al contrario in un'ottica di struttura consolidata, un disservizio grave su di un server che ospita decine e decine di applicazioni ha una portata ben diversa e provoca danni maggiori.

Anche per strutture virtualizzate esistono soluzioni per garantire alta affidabilità e tolleranza ai guasti: analogamente a ciò che avviene su macchine fisiche, è possibile ad esempio mantenere server di riserva, uguali per caratteristiche a quelli che stanno erogando i servizi, pronti ad entrare in funzione ed a sostituire l'eventuale macchina fisica non raggiungibile; questo però significa duplicare o quasi l'investimento necessario per l'acquisto delle macchine adatte a supportare il consolidamento.

1.3.3 Problemi di licenze e di supporto

Va senz'altro considerata la possibilità che alcune applicazioni non siano virtualizzabili: questo può accadere perché sono coinvolti dispositivi non comuni che non è possibile virtualizzare, oppure più frequentemente perché il produttore di software ha espressamente dichiarato che non viene fornita assistenza in caso di utilizzo della propria applicazione su sistema virtualizzato; in questo caso è necessario valutare attentamente se vale la pena accettare il rischio di rimanere senza assistenza in caso di problemi.

Altra situazione che conviene prevenire, ove possibile, consiste nell'insorgere di problemi relativi all'uso delle licenze ed alle polizze dei contratti con gli *OEM*. Può ad esempio accadere di aver acquistato licenze per un certo

numero di processori, con un contratto che fornisce servizio di assistenza per ognuno di essi per un certo numero di anni, e che in seguito al processo di consolidamento si sia ridotto il numero di licenze effettivamente necessarie: se non espressamente previsto da contratto, non è scontato il fatto che cancellando le licenze in eccesso si ottenga anche una riduzione del costo di assistenza, in quanto esso era stato inizialmente calcolato sul numero totale di licenze. A causa di questo problema si potrebbe quindi non ottenere alcun risparmio economico, e nel caso peggiore anche un aumento dei costi; si legga [8] per maggiori chiarimenti.

1.4 Alcuni casi reali di strutture consolidate

Di seguito vengono forniti i risultati di soluzioni di consolidamento applicati ad alcuni casi reali, in modo da dipingere un quadro concreto di cosa si riesca ad ottenere al termine del processo.

Dipartimento delle Risorse Idriche, Stato della California

A seguito di un progetto di consolidamento durato circa 16 mesi e partito a fine 2009, il dipartimento ha virtualizzato il 95% delle applicazioni in uso e consolidato 600 server rack, distribuiti su diverse infrastrutture, in 160 riuniti in un'unica struttura. Questo ha permesso di evitare 2.2 milioni di dollari in spese di gestione, riducendole complessivamente del 25%; grazie alla nuova struttura consolidata sono inoltre stati ridotti i consumi energetici del 40% ed è aumentata l'efficienza di raffreddamento del 50% [14].

Centro Servizi Informatici di Ateneo (Ce.S.I.A.) di Bologna

Tra il 2004 ed il 2006 è stato effettuato un processo di consolidamento applicativo e hardware sulla struttura che gestisce i servizi di Ateneo. Tale struttura era inizialmente composta da 199 server, i quali in gran parte dichiaravano in media un uso di CPU e RAM rispettivamente inferiori al 30% e 50% (vedi figura 1.1). Il consolidamento ha portato alla dismissione di una

parte di server ed alla virtualizzazione della gran parte di essi: si è passati da 199 macchine fisiche a 7 macchine fisiche (AMD Opteron a 4 core, con 16GB RAM, 6 Terabyte di SAN in dischi FibreChannel), consolidando in media su ognuna di esse 30 macchine virtuali e sfruttando in modo ottimale le risorse condivise [2].

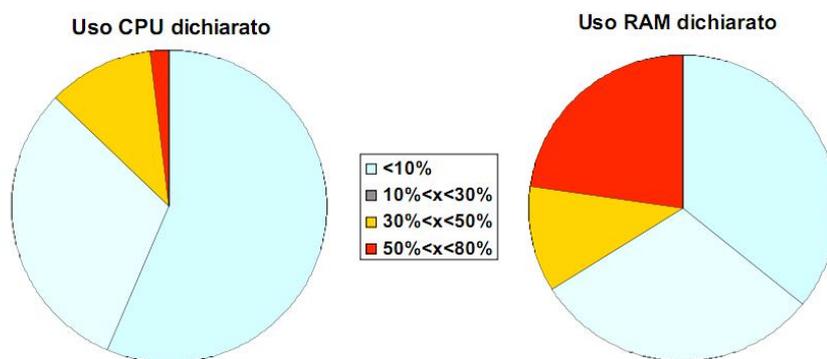


Figura 1.1: Analisi dell'utilizzo dei server prima del consolidamento, campione del 78%. Fonte: [2]

Duke Cancer Institute, North Carolina

Red Hat ha fornito supporto al processo di consolidamento nella struttura informatica del Duke Cancer Institute, con il risultato che circa il 70% degli 85 server che compongono la struttura sono stati virtualizzati. In particolare sono stati inizialmente utilizzati 5 server a 2 socket e 6 core, ognuno ospitante 11 macchine virtuali, supportati da una SAN in *Raid 6* della capacità di 96 terabyte.

Il processo si è articolato in una prima fase di migrazione delle macchine fisiche verso le macchine virtuali, attuata sfruttando il software *Acronis Backup & Recovery Virtual Edition*, quindi è seguito il loro consolidamento su piattaforma *Red Hat Enterprise Virtualization*. I vantaggi ottenuti: riduzione

delle spese di gestione, maggiore semplicità di amministrazione delle macchine ed in generale un miglioramento della qualità del servizio con maggiore stabilità e affidabilità delle applicazioni [9].

Capitolo 2

Tecniche e soluzioni per il consolidamento

Come si è già avuto modo di anticipare, la tecnica principale su cui si basa il consolidamento di server è la *virtualizzazione* [18], che secondo una delle tante definizioni consiste nella separazione di risorse e servizi dai meccanismi con cui essi vengono fisicamente serviti. Nel mondo dell'informatica questo campo si è dimostrato terreno fertile che ha avuto applicazione in diversi ambiti. Ad esempio:

- a livello di processo, si utilizzano macchine virtuali per supportare l'esecuzione di programmi. Questo avviene ad esempio per *Java*, la cui portabilità è garantita dalla macchina virtuale sottostante che lo interpreta: esso infatti viene prima compilato in *bytecode* e poi interpretato dalla *Java Virtual Machine*.
- si virtualizzano i componenti di un calcolatore: è quello che avviene ad esempio per la memoria virtuale o per i sistemi di *storage*.
- a livello di sistema, si virtualizza una macchina nella sua interezza consentendo al calcolatore fisico (da qui in avanti, macchina *host*) di contenere al suo interno una o più macchine virtuali (da qui in avanti, macchina *guest*), con i loro sistemi operativi, servizi e applicazioni.

- si virtualizzano le reti che connettono tra di loro i sistemi virtualizzati, anche se sono ospitati su host fisicamente distinti [23].

Per quanto riguarda le caratteristiche delle macchine virtuali e le proprietà che devono possedere, già nel 1974 sono stati formalizzati [15] alcuni requisiti, tutt'ora attuali e rispettati, che devono essere mantenuti nella progettazioni di architetture virtuali:

Fedeltà: Un programma in esecuzione in una struttura virtualizzata deve comportarsi sostanzialmente allo stesso modo in cui si comporterebbe se eseguito direttamente sulla macchina fisica; le uniche due eccezioni devono riguardare differenze causate dalla disponibilità di risorse di sistema oppure da tempi di esecuzione differenti.

Efficienza: La maggior parte delle istruzioni del processore virtuale devono essere eseguite direttamente dal processore reale, senza alcun intervento del software di controllo (*Virtual Machine Monitor*); in particolare tutte le istruzioni innocue devono poter essere eseguite direttamente.

Controllo delle risorse: Un programma non deve avere la possibilità di accedere direttamente alla risorse del sistema, se non a quelle allocate per il VMM; questi deve anche avere la possibilità, sotto alcune condizioni, di riprendere il controllo di risorse già allocate al programma.

Astraendo dall'effettiva realizzazione, la visione che si ottiene utilizzando una struttura con macchine virtuali è rappresentabile come da figura 2.1. Non vi è però un'unica soluzione per virtualizzare: a seconda del livello a cui viene posizionato l'*hypervisor* (il componente di sistema che realizza la virtualizzazione, chiamato anche *Virtual Machine Manager*), e del modo in cui si forniscono al sistema operativo guest i componenti necessari per essere eseguito, si ottengono realizzazioni differenti.

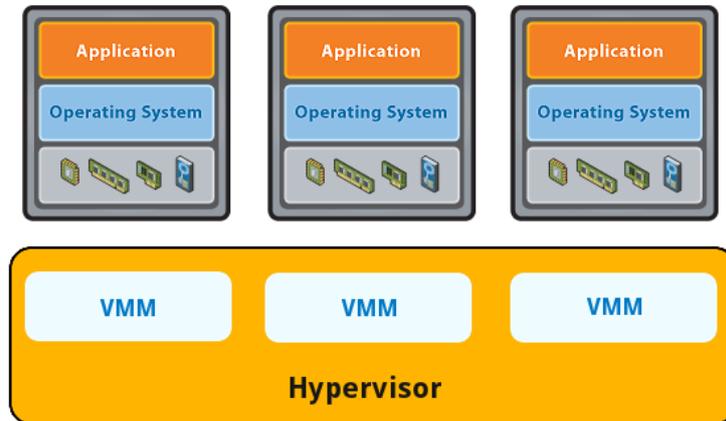


Figura 2.1: La struttura generica di un sistema con macchine virtuali

2.1 Bare-metal hypervisor (Type 1)

Un hypervisor di tipo nativo si trova in un livello immediatamente superiore a quello dell'hardware della macchina: questi è a tutti gli effetti il “vero kernel” che la gestisce (figura 2.2). Il sistema operativo guest crede di essere in modalità *kernel mode*, mentre in realtà sta girando in *user mode*: è l'hypervisor che esegue le istruzioni privilegiate oppure simula quello che farebbe l'hardware per conto del sistema guest. Essendo interfacciato direttamente con l'hardware, un hypervisor di questo tipo fornisce prestazioni superiori rispetto ad un hypervisor ospitato.

2.2 Hosted hypervisor (Type 2)

Un hypervisor di questo tipo si trova ospitato dal sistema operativo che controlla realmente la macchina (figura 2.3). L'aggiunta di un livello porta ad un aumento della complessità, poiché ad esempio ogni system call del sistema guest deve passare attraverso il proprio VMM, quindi attraverso l'hypervisor ed infine giunge al sistema operativo host. Una struttura di

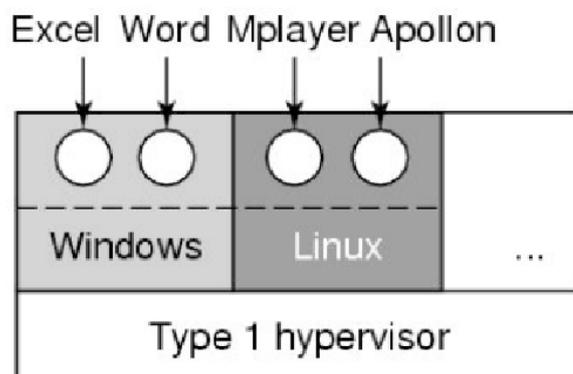


Figura 2.2: Hypervisor di tipo bare-metal (nativo)

questo tipo comporta anche di dover realizzare una versione dell'hypervisor per ogni sistema operativo su cui lo si vuole installare.

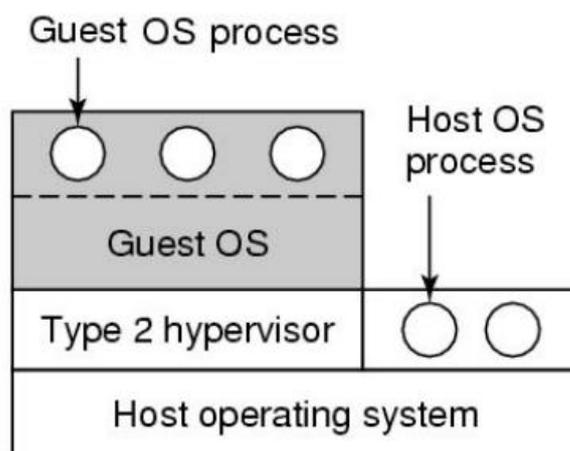


Figura 2.3: Hypervisor di tipo hosted (ospitato)

2.3 Istruzioni privilegiate e trap and emulate

In ogni architettura esistono livelli di privilegio utilizzati per vincolare l'esecuzione delle istruzioni soltanto nel livello appropriato: il sistema operativo si trova ad un livello di privilegio alto, mentre i processi utente ad un livello basso e per ovvi motivi le istruzioni di basso livello (cioè che sono dirette all'hardware) possono essere eseguite soltanto nel livello di privilegio massimo. In una struttura senza virtualizzazione infatti, il sistema operativo occupa il livello di privilegio massimo e si trova direttamente a contatto con l'hardware (figura 2.4).

In una struttura virtualizzata invece, il sistema operativo guest si trova al di sopra dell'hypervisor e di conseguenza tutte quelle istruzioni di basso livello che normalmente verrebbero eseguite, falliscono. A livello teorico sarebbe sufficiente che l'hypervisor intercettasse tali fallimenti, li eseguisse autonomamente e li facesse risultare come se fossero stati realmente eseguiti sull'hardware: una tecnica di questo tipo prende il nome di *trap and emulate*, la quale fondamentalmente cattura le eccezioni sollevate dall'esecuzione di istruzioni privilegiate e le emula sfruttando il sistema host.

In effetti, a partire dagli anni settanta, tale tecnica è stata a lungo l'unica concretamente sfruttata per virtualizzare. Con la diffusione dell'architettura x86 ci si è però accorti che virtualizzarla mediante trap and emulate comportava due grossi problemi:

1. È inefficiente in quanto ogni trap and emulate causa un *context switch*, e tutte le istruzioni privilegiate daranno origine a trap
2. La semantica di alcune istruzioni per x86 prevede un comportamento diverso a seconda che siano eseguite in user o kernel mode, quindi non sempre scatenano trap. Ad esempio l'istruzione *popf* eseguita in user mode modifica i flag della *ALU*, mentre in kernel mode modifica anche i flag di sistema [17].

È stato quindi necessario sviluppare tecniche alternative per risolvere i problemi sopra citati, quali *Full Virtualization*, *Paravirtualization* e *Hardware-assisted Virtualization*.

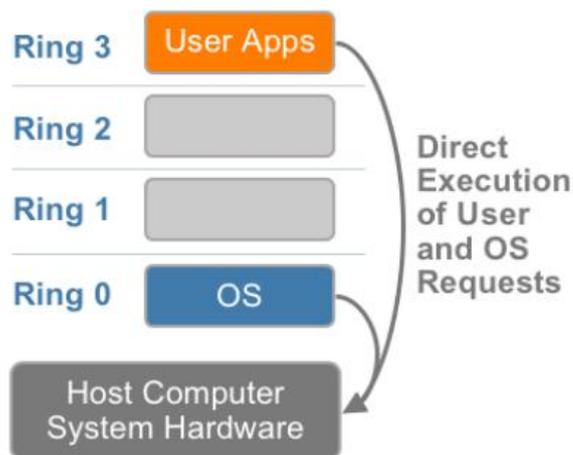


Figura 2.4: Distribuzione dei livelli di privilegio in situazione non virtualizzata

2.4 Full virtualization

La virtualizzazione totale si basa su una tecnica chiamata traduzione binaria (*binary translation*): le istruzioni non virtualizzabili sono sostituite da sequenze di istruzioni “compatibili” che possano invece essere eseguite. La traduzione, nonostante sia lenta e molto complessa, è comunque più efficiente di una soluzione basata sull’interpretazione: dover eseguire un ciclo di *fetch-decode-execute* potrebbe costare centinaia di istruzioni fisiche per ogni istruzione del sistema guest, mentre il processo di traduzione costa un *overhead* che, se gestito con tecniche di *caching*, rimane solo iniziale. Al contrario, le istruzioni non privilegiate vengono eseguite direttamente senza intaccarne le prestazioni.

Inoltre con questa tecnica il sistema operativo guest non è consapevole di essere virtualizzato e non necessita di alcuna modifica, in quanto ogni VMM fornisce tutti i servizi tipici di un sistema reale: un *BIOS* virtualizzato, dispositivi virtualizzati ecc.

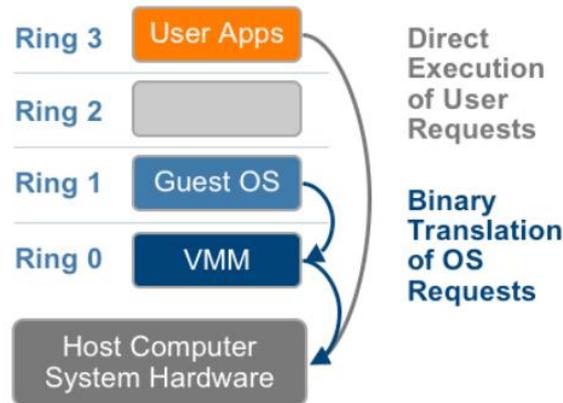


Figura 2.5: Macchina virtuale realizzata con virtualizzazione totale

2.5 Paravirtualization

Se si desidera evitare l'overhead necessario a tradurre le istruzioni privilegiate del sistema guest, l'alternativa consiste nel modificare direttamente il suo codice in modo che non contenga più quel tipo di istruzioni. Si definisce un sottoinsieme di istruzioni di x86 che siano virtualizzabili, quindi si effettua il *porting* del sistema operativo guest su quel nuovo set di istruzioni; a questo punto si ottiene codice che può essere direttamente interfacciato con l'hypervisor, senza alcun tipo di overhead aggiuntivo.

Il vantaggio principale consiste nel fatto che si elimina l'overhead necessario ad interpretare o tradurre le istruzioni non virtualizzabili; inoltre l'hypervisor può fornire chiamate di sistema dedicate per la comunicazione tra sistema operativo guest e hypervisor stesso, aumentando ancora l'efficienza. Il rovescio della medaglia riguarda ovviamente la portabilità: per poter

sfruttare questo meccanismo è necessaria la modifica del codice del sistema operativo che si desidera virtualizzare, cosa che può anche risultare impossibile se in particolare si tratta di *software proprietario* oppure se dipende da una qualche funzione che non è possibile virtualizzare.

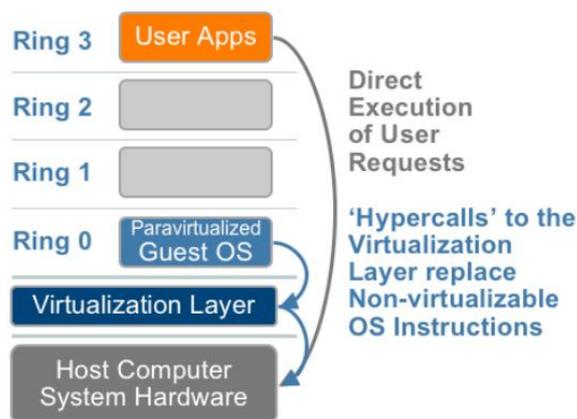


Figura 2.6: Macchina virtuale realizzata con paravirtualizzazione

2.5.1 Xen

Xen è un esempio di hypervisor basato su paravirtualizzazione, realizzato dal *Computer Laboratory* della *University of Cambridge* e successivamente mantenuto da una community che lo rilascia sotto licenza GPL2, nel quale le macchine guest vengono realizzate in strutture chiamate *domain*. Di questi ne esiste uno, chiamato *Domain0*, il quale contiene una macchina virtuale privilegiata che ha accesso all'interfaccia di controllo che gestisce la divisione delle risorse dell'host.

Da precisare che inizialmente la paravirtualizzazione era l'unico mezzo per ospitare macchine guest; a partire invece dall'introduzione delle istruzioni hardware orientate alla virtualizzazione, è stata resa disponibile la possibilità di ospitare anche macchine sfruttando full virtualization. Da notare che comunque la modifica del sistema operativo guest non si rivela partico-

lamente onerosa: per quanto riguarda Linux si tratta ad esempio di un'aggiunta di 2995 linee di codice, pari all'1.36% del codice del kernel relativo all'architettura x86 [3].

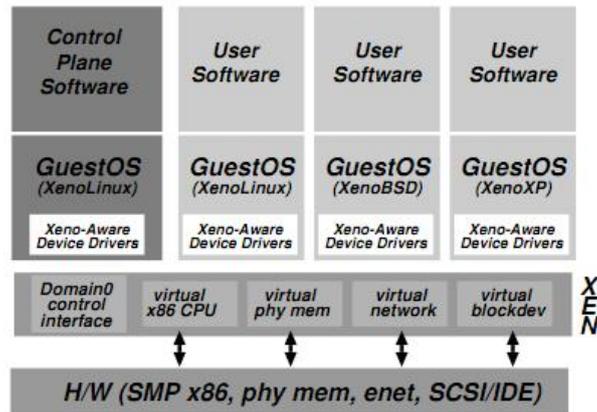


Figura 2.7: Come appare la struttura di un'architettura che utilizza Xen per virtualizzare

2.6 Hardware-assisted Virtualization

L'ultima soluzione da prendere in considerazione ha a che fare con innovazioni hardware che i produttori di processori hanno compiuto: a partire dal 2006, Intel e AMD hanno dotato le loro CPU di istruzioni dedicate per la virtualizzazione, introducendo un nuovo livello di esecuzione del processore ai quattro già esistenti; questo permette all'hypervisor di lavorare al di sotto del *Ring 0* contenente il sistema operativo guest.

In questo modo non vi è alcun bisogno di effettuare traduzione binaria o paravirtualizzazione, in quanto le istruzioni privilegiate vengono direttamente catturate dall'hypervisor con un meccanismo trap and emulate di cui si è già parlato. Senza entrare nei dettagli, in pratica viene utilizzata una nuova modalità di esecuzione chiamata *guest mode*, in cui si entra tramite l'istruzione *vmrun* e si esce con l'istruzione duale; al suo interno le istruzioni

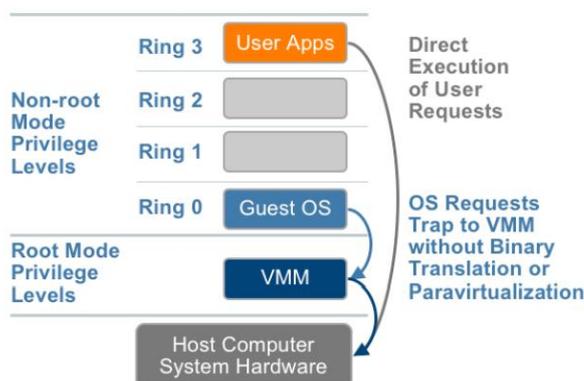


Figura 2.8: Macchina virtuale realizzata con virtualizzazione assistita dall'hardware

del sistema operativo guest, comprese quelle privilegiate, vengono eseguite come se esso fosse nativo e quindi praticamente senza alcun overhead. Si deduce che le prestazioni di questa soluzione dipendono fondamentalmente dal numero di uscite dalla guest mode che vengono effettuate, cosa che avviene quando è necessario eseguire ad esempio operazioni di *I/O*.

2.6.1 KVM

Kernel-based Virtual Machine [16] è un modulo open source caricabile su kernel Linux, il quale consente al sistema operativo di diventare un hypervisor bare-metal. KVM si distingue in particolare per due caratteristiche:

1. è stato sviluppato dopo l'introduzione della virtualizzazione hardware citata prima, quindi fa utilizzo delle funzionalità di virtualizzazione realizzate via hardware senza doverle realizzare via software.
2. non contiene al suo interno un'implementare di molti dei componenti di cui ha bisogno un hypervisor (scheduler dei processi, gestore della memoria ecc), in quanto essendo caricato sul kernel di Linux può sfruttare funzionalità che esistono già.

Questo porta KVM ad essere un sistema di virtualizzazione molto leggero, sia dal punto di vista del numero di linee di codice sia per quanto riguarda la sua struttura: è sostanzialmente composto da un *device driver* che gestisce la virtualizzazione hardware e da una versione leggermente modificata di Qemu, un emulatore di processore caratterizzato da un veloce meccanismo di traduzione dinamica [4]. In questo modo le macchine virtuali sono a tutti gli effetti normali processi di un sistema Linux, che traggono vantaggio dalla stabilità e dal livello di sicurezza che il kernel offre.

Inoltre KVM supporta la possibilità di paravirtualizzare la gestione dell'I/O dei sistemi guest: utilizzando i driver *virtio* si possono ottenere performance maggiori sulle operazioni su disco e su rete, traendo vantaggio dai benefici della paravirtualizzazione.

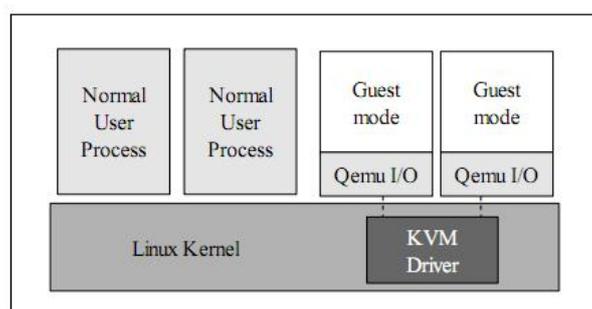


Figura 2.9: Come appare la struttura di un'architettura che utilizza KVM per virtualizzare

2.7 Consolidamento e outsourcing: Cloud computing

Come già spiegato in precedenza, utilizzare tecniche di virtualizzazione permette di consolidare un'infrastruttura informatica ottenendo risparmio in risorse, energetiche ed economiche, ed un miglioramento in efficienza; fin'ora si è dato per scontato che la loro attuazione pratica avvenisse sempre *in house*

all'organizzazione, cioè all'interno delle proprie strutture ed utilizzando le proprie risorse.

In realtà questa non è l'unica via percorribile: è possibile anche utilizzare soluzioni in *outsourcing*, affidandosi cioè ad aziende e società esterne, le quali erogano il servizio desiderato al posto dell'organizzazione stessa. In particolare, il *Nation Institute of Standards and Technology* ha pubblicato la seguente definizione di *Cloud computing* [13]:

“Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

Va precisato che cloud computing e virtualizzazione non sono la stessa cosa: il primo termine indica fundamentalmente un modello di business, mentre il secondo una vera e propria tecnologia. A livello teorico effettuare cloud computing non implica utilizzare tecnologie di virtualizzazione, e viceversa virtualizzare non obbliga a portare i servizi “nella nuvola”: allo stesso tempo è però innegabile che il modo più naturale per implementare le funzionalità tipiche del cloud computing, passa proprio per la virtualizzazione [21, 11].

2.8 Caratteristiche

Sono stati individuati tre diversi livelli di servizi di Cloud:

Software as a Service (SaaS): quando si forniscono applicazioni ospitate nella nuvola e gestite dal provider: il caso forse più noto è il servizio *Gmail* offerto da Google.

Platform as a Service (PaaS): quando si dà la possibilità al consumatore di installare nella nuvola le proprie applicazioni, sviluppate con linguaggi di programmazione supportati dal provider; *Google App Engine* e *Windows Azure* sono due esempi.

Infrastructure as a Service (IaaS): quando si permette al consumatore di gestire direttamente una parte di infrastruttura, consentendo accesso a sistema operativo, sistemi di storage, componenti di rete ecc. *Amazon Web Services* fornisce ad esempio questo tipo di servizio.

Vi sono anche tre principali modelli di implementazione di Cloud:

Public Cloud: la struttura è di proprietà di un soggetto (provider) che la gestisce e permette l'accesso ai clienti con modalità *pay-as-you-go*.

Private Cloud: la struttura si trova all'interno dell'organizzazione, che la gestisce e la utilizza per servizi personali.

Hybrid Cloud: la struttura è realizzata come unione di caratteristiche appartenenti a modelli diversi.

2.9 Consolidare nella nuvola

Quali possibilità offre una soluzione basata su Cloud, dal punto di vista dell'organizzazione che desidera consolidare i server della propria struttura? Come accade spesso, la scelta si riduce ad un *trade-off*: vi sono alcuni vantaggi, soprattutto dal punto di vista economico, ma anche problemi strutturali che nel caso peggiore possono portare ad una perdita economica [1].

2.9.1 Elasticità e provisioning rapido

Delegare la gestione dei server ad una società esterna può consentire di tagliare la spesa necessaria per l'acquisto e la gestione di gran parte dell'hardware, permettendo inoltre di ridurre il numero di tecnici che normalmente avrebbe amministrato grandi parchi macchine.

Pagare inoltre secondo *server-hours* permette di decidere volta per volta di quante risorse si necessita, consentendo di scalare verso l'alto o verso il basso con tempi decisamente più ridotti di quanto richiederebbe una struttura fisica. Questo ha importanza in particolare per la gestione dei picchi di

richieste, in quanto un modello di pagamento basato sul consumo si adatta meglio a quel genere di eventi poco prevedibili: si pensi ad esempio alla messa in funzione di un nuovo servizio di cui non si può prevedere il successo (e quindi la quantità di risorse necessarie), oppure il dover rispondere ad un evento improvviso che sta generando traffico elevato o, al contrario, un calo di utilizzo della propria applicazione; il fattore elasticità consente quindi di adattarsi meglio alla dinamicità intrinseca di certi fenomeni.

2.9.2 Lock-in e riservatezza

Una soluzione in outsourcing comporta sempre una forma di dipendenza più o meno elevata: cosa succede se coloro ai quali ci si è affidati non sono più in grado di fornire il servizio richiesto? Oppure se con il passare del tempo essi modificano i termini di contratto, aumentano i costi, introducono caratteristiche non gradite? Dipendere da un'entità esterna per un servizio così critico, come può essere l'infrastruttura informatica, può rivelarsi una scelta poco saggia: un guasto sul Cloud provider può ad esempio trasformarlo in *single point of failure* per l'organizzazione che ci si è affidata.

Inoltre all'interno del Cloud vengono utilizzate API tipicamente proprietarie, in quanto non esiste ancora un vero e proprio standard; questo porta ad esempio all'impossibilità da parte del consumatore di esportare i propri dati su altri provider, creando di fatto un effetto di dipendenza che può rivelarsi pericoloso: cosa succede se il provider fallisce?

Infine, va anche valutato il problema della riservatezza dei dati: una Public Cloud per definizione non appare una scelta adatta per il trattamento di informazioni riservate, anche in presenza di crittografia, in quanto dal punto di vista del cliente essa appare come una vera e propria scatola nera, di cui spesso non si conosce neppure la locazione fisica di CPU, RAM e storage che si stanno utilizzando. Questi problemi sconfinano poi in questioni etiche se il cliente in questione opera con informazioni private dei cittadini, come ad esempio una Pubblica Amministrazione.

Capitolo 3

Analisi delle fasi del processo

Dopo aver parlato di *cosa* sia il consolidamento e di *quali tecniche* si utilizzino per consolidare più macchine in un'unica macchina, si può passare ad analizzare il problema affrontandolo dal punto di vista di chi si trova a dover seguire il processo di consolidamento in una infrastruttura informatica. Fornendo un modello di approccio al consolidamento organizzato in fasi distinte, ognuna delle quali caratterizzata da un certo tipo di operazioni da eseguire, problemi comuni e strategie tipiche per scegliere la via migliore, è possibile tentare di dare risposta a quesiti come i seguenti:

- In che modo decido quali sono i server adatti ad essere consolidati?
- In base a quali fattori si scelgono i server che dovranno condividere la stessa macchina fisica?
- Come analizzo lo stato della mia struttura informatica e ricavo i dati necessari?
- È meglio consolidare sullo stesso server i servizi simili o i servizi diversi?

3.1 Definizione degli obiettivi

Il punto di partenza è definire gli obiettivi che si vuole raggiungere, a partire dalla valutazione di quali sono i limiti della propria infrastruttura a

cui si vuole porre rimedio. Si può agire per rimediare a:

Limiti strutturali: come il termine di spazio fisico nel datacenter o il raggiungimento del limite di risorse elettriche disponibili. Si noti che in alcuni casi effettuare un ulteriore upgrade delle strutture fisiche quali l'impianto elettrico, risulterebbe in una spesa enorme.

Limiti economici: quando ad esempio il budget limitato impedisce di acquistare hardware dedicato per la copertura di nuovi progetti e per la loro gestione.

Limiti operativi o tecnologici: se il funzionamento stesso dell'infrastruttura pone limiti sulla possibilità di ottenere nuove funzionalità e profitti.

3.1.1 Approccio di contenimento

Se si vuole soltanto rimediare alla proliferazione di server, in modo da poter continuare a realizzare nuovi progetti senza superare i limiti di budget, recuperare spazio fisico e risparmiare in energia elettrica, probabilmente è sufficiente adottare una soluzione di *contenimento*: questo consente di ottenere ottimi risultati senza dover stravolgere completamente l'infrastruttura informatica.

Forward containment

Se tutti i nuovi progetti vengono gestiti su macchine virtuali, si ferma la crescita del numero di server e soprattutto si rimanda la spesa necessaria per l'acquisto di hardware. Anche nel caso in cui non si possano riutilizzare server già acquistati per gestire le macchine virtuali e si debba acquistarne di nuovi, il costo viene recuperato piuttosto rapidamente grazie anche al risparmio in costi di gestione.

L'isolamento e la portabilità che una macchina virtuale garantisce, consentono inoltre di amministrare più facilmente il software in fase di sviluppo:

lo si può testare su sistemi operativi diversi con tempi di provisioning nell'ordine dei minuti, evitando inoltre che eventuali malfunzionamenti danneggino il resto del sistema.

Backward containment

Specularmente, si può agire per trasferire le applicazioni legacy all'interno di macchine virtuali su server recenti: questo consente di ritirare l'hardware obsoleto che veniva mantenuto soltanto per poter far girare applicazioni di quel tipo. In questo modo si limita da un lato i possibili malfunzionamenti che derivano dall'utilizzo di software non più supportato, e dall'altro si elimina il problema della mancanza di pezzi di ricambio per hardware non più in commercio.

3.1.2 Approccio strutturale

Se si desidera invece consolidare l'intera struttura è necessario un approccio più ragionato, soprattutto se si parla di un'organizzazione con centinaia di macchine. Una buona idea potrebbe essere identificarne un piccolo sottoinsieme, il più possibile simile all'infrastruttura nel suo complesso, ed eseguire un "consolidamento pilota" su di esso.

In questo modo si ha la possibilità di sperimentare su piccola scala parte dei problemi che probabilmente si dovranno affrontare in grande, permettendo nel contempo di ottenere una piccola conquista che può anche essere sfruttata per pubblicizzare e promuovere un intervento su larga scala.

3.2 Raccolta dati e analisi workload

Le scelte di consolidamento vengono fatte in seguito all'analisi dei *workload*, ossia del carico di lavoro - inteso come consumo di risorse del server - a cui ogni servizio è sottoposto. Per ricavare queste e altre informazioni è nec-

essaria una fase di analisi dei server che può essere condotta, nel caso non si stiano già utilizzando servizi appositi, con adeguati software di monitoraggio.

3.2.1 Monitoring tools

Un software di monitoraggio, come ad esempio *Ganglia* [12], consente di avere sotto controllo lo stato delle macchine e la quantità di risorse che esse stanno utilizzando, attività che senza strumenti automatici può rivelarsi particolarmente gravosa se si lavora con un parco macchine di ampie dimensioni. Strumenti open source di questo tipo possono essere utilizzati anche per la raccolta dati necessaria ad effettuare scelte di consolidamento, offrendo un'alternativa ai sistemi di *planning* forniti da alcuni produttori, i quali tipicamente hanno il difetto di poter essere usati soltanto congiuntamente agli altri software del pacchetto. *Ganglia* permette ad esempio, tra le varie funzionalità, di definire metriche personalizzate e di ricavarne i grafici corrispondenti; monitorando il cambiamento nel tempo di metriche personalizzate e metriche classiche, quali ad esempio consumo di CPU, RAM occupata, quantità di I/O effettuato su rete ecc, si ricavano una quantità notevole di informazioni che possono guidare nelle scelte di consolidamento.

Informazioni da ricavare

Utilizzando strumenti come quelli citati sopra si suggerisce di ricavare una serie di documenti di questo tipo [20]:

Inventario delle risorse: un catalogo di tutte le macchine attualmente disponibili, elencando per ognuna le caratteristiche tecniche e le informazioni sul numero di applicazioni e servizi attivi.

Tabelle delle tipologie di workload e classi di risorse utilizzate: una serie di tabelle in cui ai workload vengano associati i valori medi di utilizzo delle canoniche classi di risorse: CPU, RAM, disco ed I/O su rete. Questo può esser effettuato a vari livelli di granularità, valutando ad

esempio i workload a livello di server, gruppi di server o applicazioni, eventualmente variando il livello di dettaglio delle risorse descritte.

È inoltre opportuno:

- effettuare la fase di analisi dei workload per un tempo sufficientemente lungo a registrare anche i momenti di picchi di richieste di risorse. Questo può voler dire anche molte settimane.
- annotare eventuali vincoli di dipendenza o incompatibilità tra applicazioni-hardware e applicazioni-applicazioni.

3.3 Progettazione della struttura

Quando si hanno dati a sufficienza che descrivano i workload delle proprie macchine, bisogna passare alla fase in cui si decide quali server consolidare e su quale macchina ognuno di essi vada consolidato; la progettazione generale della nuova infrastruttura deve chiaramente tener conto di tutte le informazioni ricavate nella fase precedente. Di seguito verranno esaminate strategie e possibilità da prendere in considerazione in questa fase.

3.3.1 Opzioni di consolidamento

Il punto centrale riguarda la scelta di quali macchine debbano essere virtualizzate e consolidate sullo stesso server. A questo proposito si possono valutare diverse opzioni:

1. raggruppare per funzionalità, cioè consolidare sulla stessa macchina fisica quelle macchine che in qualche modo erano legate in quanto funzioni diverse di un unico sistema.
2. raggruppare per performance, cioè consolidare sulla stessa macchina fisica una serie di macchine che consentano di massimizzare l'utilizzo delle risorse.

3. consolidare su stessa macchina i workload con carichi temporalmente complementari.
4. consolidare su stessa macchina i workload identici.

Approccio per funzionalità

Un approccio per funzionalità sacrifica il guadagno in termini di utilizzo di risorse in favore di una gestione più facile delle macchine virtuali, in quanto centralizzate sullo stesso server; questo porta anche vantaggi dal punto di vista della sicurezza dell'intera struttura, dato che si concentrano diversi possibili bersagli su di una singola macchina.

Un esempio in questo senso è la realizzazione di una *DMZ* (De Militarized Zone) consolidata sullo stesso server: come si vede in figura 3.1 è possibile utilizzare due schede di rete fisiche per instradare il traffico al livello di virtualizzazione, dove switch virtuali a loro volta lo instradano verso i server di frontiera che fungono da firewall (installati su macchine virtuali). Questi possono quindi consentire o meno il passaggio del traffico verso web e application server (anche essi realizzati su macchine virtuali). L'isolamento e la stratificazione su livelli diversi aumentano la sicurezza del sistema.

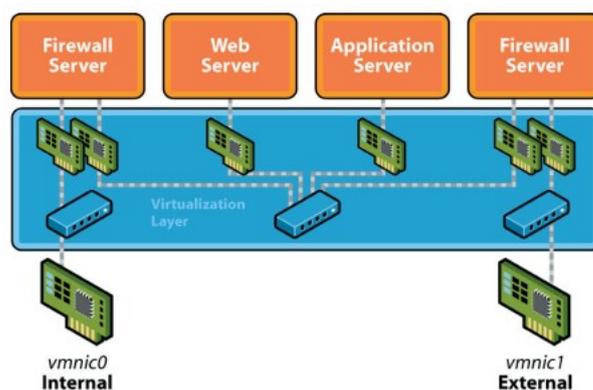


Figura 3.1: Realizzazione di una DMZ consolidata su di un unico server fisico.
Fonte: [22]

Approccio per performance

Al contrario, un approccio per performance verte sul tentativo di sfruttare nel modo migliore le risorse dei server fisici, virtualizzando su di essi il maggior numero possibile di macchine. In questo modo si ottiene un utilizzo ottimale delle risorse, a scapito di possibili complicazioni a livello di struttura logica del sistema, dato che sullo stesso server potrebbero anche essere consolidati applicazioni e servizi totalmente diversi tra di loro.

Si noti che problemi di ottimizzazione di questo tipo sono oggetto di studi, ad esempio nella forma di *Static Server Allocation Problem* [19] e *Multi-Objective Multi-Dimension Bin Packing Problem* [10], e si è dimostrato che si tratta di problemi *NP-hard*.

Workload complementari

Altro aspetto che va valutato è la distribuzione temporale dei workload. Alcuni servizi per loro natura tendono ad impegnare le risorse in momenti diversi della giornata: ad esempio i servizi di backup vengono verosimilmente effettuati durante la notte, i server email al contrario potrebbero avere picchi al mattino quando il personale entra in ufficio, così come servizi di *LDAP*. Questo significa che servizi con workload complementari sono buoni candidati ad essere consolidati sullo stesso server, in quanto occuperebbero le risorse in momenti diversi della giornata (figura 3.2).

Workload identici

Al contrario di un consolidamento per workload complementari, consolidare lo stesso tipo di workload non massimizza evidentemente l'utilizzo delle risorse, dato che i picchi avverranno negli stessi momenti; inoltre ci si potrebbe chiedere se sia saggio concentrare tutti i servizi di un dato tipo su di uno stesso server, con il rischio che un eventuale guasto hardware possa metterli offline tutti in una volta. Se però si ha a che fare con servizi che utilizzano una gran quantità di memoria su disco, è possibile ottenere un

risparmio di memoria utilizzata facendo in modo che parte dei file necessari siano comuni tra le varie istanze, e non replicati come avverrebbe se fossero ospitati su macchine fisicamente distinte. Inoltre si potrebbe ottenere un risparmio in spese di licenze, dal momento che molti produttori associano il numero di licenze da acquistare al numero di computer fisici che utilizzano l'applicazione, anche se sono sempre di più le aziende che hanno aggiornato i propri contratti prevedendo il caso di un'installazione su macchina virtuale.

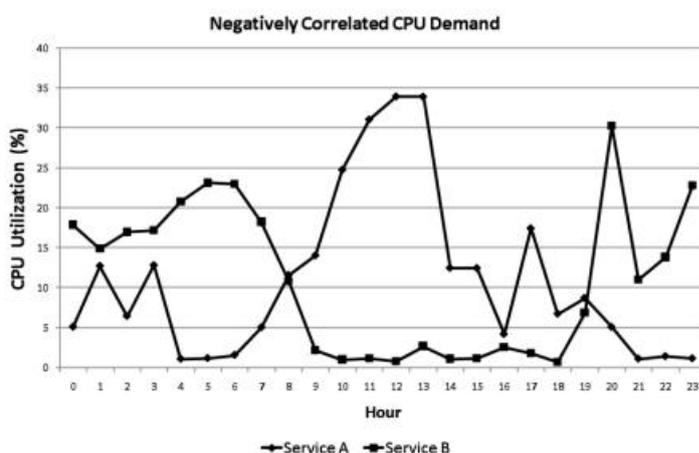


Figura 3.2: Due servizi i cui workload impiegano risorse in momenti complementari durante la giornata. Fonte: [19]

3.3.2 Overhead della virtualizzazione

Un fattore importante che va tenuto presente riguarda l'overhead in termini di risorse che aggiunge il livello di virtualizzazione. È necessario in particolare valutare

1. il consumo di CPU aggiuntivo che genera la macchina virtuale su cui è consolidato un sistema.
2. in che modo i componenti del livello di virtualizzazione impattano sulle performance delle applicazioni ospitate.

In generale l'overhead dipende dal tipo di virtualizzazione realizzata e da quale strumento si utilizza, anche se spesso esiste un legame di proporzionalità diretta tra overhead della CPU e quantità di I/O effettuato [6]; questo in particolare nel sistema Xen in cui le operazioni di I/O delle macchine guest vengono effettuate dal driver del Domain0.

Esistono tuttavia modelli [25] che consentono di prevedere la quantità di risorse richieste da qualsiasi applicazione, realizzati a partire da *microbenchmark* eseguiti sul sistema fisico e sul sistema virtuale: tramite di essi è possibile ricavare la relazione che lega il consumo di risorse in ambiente nativo e ambiente virtuale. Considerazioni di questo tipo sono chiaramente importanti per capire quante risorse destinare alla macchina fisica che ospiterà i server guest.

3.3.3 Best practices

Oltre alle soluzioni e tipologie di approcci citati precedentemente, si possono definire alcune regole che in generale vanno considerate:

1. se due o più applicazioni entrano in conflitto, ad esempio perché utilizzano la stessa porta oppure a causa delle rispettive strategie di sicurezza, è necessario evitare la situazione ed eventualmente consolidarle su server differenti.
2. la somma delle risorse richieste per i workload di ogni servizio virtualizzato non può in nessun istante essere maggiore delle capacità della macchina host.
3. se si consolidano sulla stessa macchina servizi con forti interazioni, come ad esempio web server e database, si aumenta la qualità del servizio; al contrario destinarli su macchine diverse aumenterebbe il tempo necessario alla trasmissione dei dati.

3.4 Migrazione verso la struttura virtuale

Dal punto di vista organizzativo migrare un numero elevato di servizi verso strutture virtuali può essere impegnativo, e a tal proposito si è anche parlato di utilizzare un approccio pilota e verificare in piccolo le difficoltà del consolidamento; a prescindere dal fatto che sia stato scelto un approccio del genere è comunque consigliabile, in questa fase, procedere alla migrazione operando di volta in volta su sottoinsiemi diversi di macchine, in accordo ovviamente con le scelte di consolidamento adottate nelle fasi precedenti. Prima di effettuare la migrazione, potrebbe inoltre essere una buona pratica la realizzazione di un documento, specifico per ogni sottoinsieme di macchine, contenente le operazioni di emergenza da eseguire per riportare un sistema in migrazione allo stato iniziale.

Dal punto di vista tecnico, invece, le operazioni da svolgere dipendono dal software di virtualizzazione che si è scelto di utilizzare: da parte dei maggiori produttori di software che operano nel campo sono state sviluppate apposite suite di programmi con le quali gestire la fase di virtualizzazione e, successivamente, di amministrazione delle nuove strutture virtuali.

Attualmente emergono in particolare le seguenti realtà:

Red Hat che produce *Red Hat Enterprise Virtualization*, composto da un hypervisor basato su KVM ed un sistema di gestione delle macchine virtualizzate.

Citrix che con *XenServer* offre sistemi di virtualizzazione basati sull'hypervisor Xen; l'azienda è inoltre molto attiva nel campo della virtualizzazione di desktop.

VMware azienda che sviluppa soluzioni di virtualizzazione bare-metal e hosted: nel primo caso opera in particolare con la suite di programmi *vSphere*.

Microsoft la quale produce Hyper-V, che al contrario delle altre principali soluzioni (di tipo bare-metal) richiede l'utilizzo di Windows Server 2008.

Senza entrare nella discussione di quale sia il prodotto migliore tra quelli citati e delle caratteristiche che ognuno di essi possiede, si porrà ora attenzione sul passo principale che è necessario effettuare in questa fase.

3.4.1 Physical to Virtual Migration

La migrazione di un sistema fisico verso un sistema virtuale (operazione chiamata spesso *p2v migration*) consiste nel trasportare ciò che prima veniva eseguito su macchina fisica in qualcosa che possa essere eseguito su macchina virtuale: occorre cioè effettuare una copia del sistema operativo, del contenuto dei dischi, delle applicazioni installate ecc ottenendo un file, normalmente chiamato *immagine del disco virtuale*, che possa essere utilizzato dalla macchina virtuale; fatto questo è necessario effettuare quelle modifiche di cui ha bisogno il nuovo sistema per poter essere lanciato, come ad esempio l'installazione dei driver adatti al nuovo hardware virtuale e delle componenti specifiche del motore di virtualizzazione scelto.

In particolare si parla di:

migrazione a caldo (hot migration o hot cloning) se la copia del server fisico avviene mentre questo è ancora in funzione.

migrazione a freddo (cold migration o cold cloning) se la copia del server fisico avviene mentre questo è offline.

Nel primo caso si ha ovviamente il vantaggio di non creare disservizi al sistema, se non un possibile calo di prestazioni dovute alle risorse utilizzate per la copia; d'altra parte è necessario gestire il fatto che possano avvenire cambiamenti nel sistema originale mentre questo viene copiato (si pensi all'attività di un database), dando origine ad incoerenze tra sorgente e destinazione. Nel secondo caso invece il processo è più semplice in quanto non esiste alcun

problema di possibili incoerenze, ma è necessario mettere il sistema offline per tutta la durata della copia.

Esistono numerosi prodotti adatti ad eseguire migrazione p2v, dotati di livelli differenti di automazione [24]: i più sofisticati sono in grado di migrare autonomamente una macchina fisica verso una virtuale, operando direttamente su rete e senza impegno dell'utente se non nella scelta di origine e destinazione, gestendo anche la migrazione a caldo.

Se si opera invece senza l'ausilio di tools dedicati, per quanto riguarda la copia dei file è conveniente utilizzare programmi come *rsync*; in particolare se si migra a caldo un host che contiene un servizio soggetto a frequenti cambiamenti, si può utilizzare il seguente protocollo [7]:

1. mentre il sistema è ancora attivo si effettua la prima copia tramite *rsync* del servizio interessato.
2. si ferma il servizio.
3. si effettua una seconda copia con *rsync*, la quale interesserà soltanto i cambiamenti avvenuti rispetto alla prima copia.
4. si avvia il sistema destinazione e si spegne quello sorgente.

Chiaramente il disservizio è tanto più evidente quanto più è ampia la quantità di tempo che passa tra i punti due e quattro citati sopra.

3.4.2 Virtual to Virtual Migration

In modo simile alla migrazione da struttura fisica a struttura virtuale, ci si può trovare a dover migrare una macchina già virtualizzata verso un altro tipo di macchina virtuale; questo se in particolare l'infrastruttura su cui si sta operando ha già subito in passato virtualizzazioni parziali, oppure se si sta consolidando verso una soluzione diversa di virtualizzazione rispetto quella attuale (ad esempio si passa da hypervisor Xen a KVM).

Anche in questo caso esistono strumenti che consentono di convertire macchine basate su hypervisor diversi (per quanto riguarda conversioni verso KVM si veda ad esempio *virt-v2v* [5]); in teoria se la virtualizzazione fosse implementata mantenendo il giusto livello di astrazione, dovrebbe essere possibile semplicemente copiare l'immagine del disco virtuale interessato verso la nuova posizione, dal momento che un disco virtuale per Xen non dovrebbe differire da un disco virtuale per KVM. Ciò è in parte vero, ma come si è visto esistono modalità di virtualizzazione che richiedono modifiche ai sistemi operativi guest. Per questo motivo, un software che si occupa di v2v migration esegue ad esempio anche le operazioni necessarie per rimuovere il software relativo all'hypervisor in sostituzione ed installare quello relativo al nuovo hypervisor.

3.5 Gestione della struttura

Terminata la fase di migrazione di tutte le macchine fisiche verso le strutture virtuali scelte, è importante dedicare una fase di monitoraggio della struttura in modo da rilevare eventuali host troppo carichi di macchine virtuali, o al contrario non sufficientemente sfruttati. È quindi necessario poter disporre di strumenti per la cosiddetta *live migration*, da non confondere con le tipologie di migrazione citate prima.

Per *live migration* si intende infatti lo spostamento di una macchina virtuale residente sull'host A all'interno dell'host B, solitamente per motivi di performance relativi ai workload. Chiaramente la tecnologia di virtualizzazione deve essere la stessa, ma è possibile spostare macchine guest residenti su host Intel verso host AMD e viceversa; inoltre si possono spostare macchine guest a 32bit sia su host a 32bit sia a 64bit, mentre ovviamente una macchina guest a 64bit può solo essere spostata su host a 64bit.

Strumenti per effettuare *live migration* sono importanti perché consentono di poter ribilanciare e ridistribuire i workload su tutte le macchine disponibili. Questo è utile in primo luogo perché consentono di adattarsi all'evoluzione

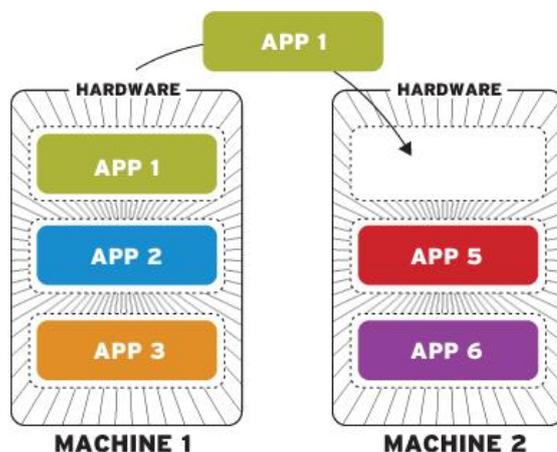


Figura 3.3: Esempio di live migration tra due host Fonte: redhat.com

dinamica dei workload nel tempo, dando la possibilità di continuare a massimizzare l'utilizzo delle risorse dei server anche successivamente al primo consolidamento; in secondo luogo perché possono essere utilizzati per gestire le situazioni di guasti: se ci si dovesse ad esempio accorgere che un server sta segnalando errori riguardanti parti critiche del sistema, oppure si dovessero registrare temperature eccessivamente alte, si potrebbero trasferire tutte le macchine guest verso altri host, in modo da prevenire i disagi che causerebbe il server in questione se andasse veramente offline.

In modo simile a quanto avviene per la migrazione fisica anche la live migration può essere svolta offline, cioè mentre il sistema interessato non è attivo, oppure a *runtime*: in questo caso gli strumenti che si occupano dell'operazione riescono praticamente ad annullare il tempo di disservizio che causa l'operazione.

Tutte le suite di programmi citate in precedenza offrono un proprio strumento per effettuare live migration, e gestire tramite interfacce grafiche il funzionamento delle macchine virtuali.

Conclusioni

Come si ha avuto modo di leggere, descrivere il tema del consolidamento di server significa descrivere una serie di aspetti diversi: quelli di carattere tecnologico e tecnico, relativi in particolare alla virtualizzazione, alle prestazioni dei server e agli strumenti necessari, e quelli relativi al management di una struttura complessa, con annessi i problemi di carattere economico e di sviluppo di un'azienda. In questa tesi si è cercato di fornire una visione organica del settore, analizzando motivazioni, tecniche, procedure e strumenti che consentono di effettuare processi di consolidamento; per quanto possibile, lo si è cercato di fare senza legarsi ad una soluzione software particolare dell'una o dell'altra azienda, se non citando alcuni casi a titolo esemplificativo. L'intento generale era dunque quello di fornire un modello che, idealmente, potesse essere seguito come una sorta di guida per effettuare il consolidamento della propria struttura; per quanto possibile, considerando anche la natura non particolarmente formalizzabile dell'argomento, si è cercato proprio di dipingere questo tipo di idea.

Il settore che lega il consolidamento di server, e più in generale la virtualizzazione intesa come strumento per semplificare e meglio amministrare le strutture in ambito *enterprise*, si tratta a mio avviso di un terreno estremamente fertile e che sarà sviluppato anche in futuro, come ad esempio nell'ambito della virtualizzazione di desktop, di cui non si ha avuto modo di discutere ma che potrebbe ridisegnare il modo tradizionale in cui un dipendente si relaziona al proprio ambiente di lavoro, il computer. Personalmente lo ritengo inoltre un ambito molto interessante anche dal punto di vista pro-

fessionale, in quanto mi ha ad esempio dato modo di confermare che business ed open source non sono affatto due mondi separati, ed anzi esistono aziende che fanno mercato nonostante vendano prodotti nati da progetti appunto open source.

Per concludere, alcune considerazioni di carattere personale. La prima cosa che ho notato durante la fase di documentazione è stata l'incredibile quantità di materiale commerciale, relativo al consolidamento, in cui sono incappato: molto spesso ho avuto difficoltà a riuscire a farmi un'idea imparziale e generale dell'argomento che stavo trattando, in quanto appunto la maggior parte del materiale trovato riguardava materiale pubblicitario sulla soluzione del produttore di turno, e non una descrizione accademica e veramente scientifica; mi sono spiegato questo fatto con l'interesse che il mercato ha attualmente in questo campo, come si sottolineava prima, e con il fatto che si tratti comunque di un settore relativamente nuovo.

Ho avuto inoltre l'impressione che si tratti di un mercato molto competitivo, nel quale vince chi fondamentalmente riesce a fornire il sistema di gestione di macchine virtuali più completo per funzionalità e più facile da gestire; in questo senso ho avuto la sensazione, in parte confermata leggendo opinioni di addetti del settore e sviluppatori, che attualmente alle soluzioni open source non manchino tanto prestazioni o funzionalità particolari di cui il cliente ha bisogno, ma manchi proprio quel componente centralizzato da cui gestire tutto, che troviamo invece nelle soluzioni commerciali. Da questo punto di vista credo ad esempio ci siano ampi margini di sviluppo.

Infine, che cosa ho imparato lavorando a questa tesi: sicuramente ho avuto modo di analizzare più in dettaglio alcune tecnologie che già conoscevo, come la virtualizzazione, soprattutto osservando l'argomento da punti di vista diversi da come li avevo affrontati fin'ora; ancora, ho avuto modo di scoprire le soluzioni e gli approcci principali per un processo di consolidamento, facendomi un'idea precisa degli attori e dei mezzi coinvolti in questo gioco; per concludere, ho avuto modo di interessarmi all'argomento ed ho scoperto che tale interesse è rimasto anche al termine della scrittura di questa tesi,

e credo quindi che anche in futuro continuerò a seguire ciò che riguarda il consolidamento di server.

Bibliografia

- [1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Andrew Katz, Randy H. and Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds: A berkeley view of cloud computing. Technical report, Electrical Engineering and Computer Sciences University of California at Berkeley, February, 2009.

- [2] Simone Balboni. Tecnologie di virtualizzazione per il consolidamento di server. <http://lia.deis.unibo.it/Courses/SistOpLS0506/materiale/Virtualizzazione.pdf>, 2006.

- [3] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *Proceedings of the nineteenth ACM symposium on Operating systems principles*, SOSP '03, pages 164–177, New York, NY, USA, 2003. ACM.

- [4] Fabrice Bellard. Qemu, a fast and portable dynamic translator. In *FREENIX Track: 2005 USENIX Annual Technical Conference*, pages 41–46. USENIX Association, 2005.

- [5] Matthew Booth. Converting, inspecting, & modifying virtual machines with red hat enterprise linux 6.1. In *Red Hat Summit 2011*. USENIX Association, 2011.

-
- [6] Ludmila Cherkasova and Rob Gardner. Measuring cpu overhead for i/o processing in the xen virtual machine monitor. pages 387–390.
- [7] OpenVZ Linux Container. Physical to container. http://wiki.openvz.org/Physical_to_VE.
- [8] Phil Downe. Virtualization and other licensing pitfalls. <http://itnegotiations.com/Clouding%20the%20Issue.pdf>, 2010.
- [9] Red Hat. Red hat provides duke cancer institute with a more cost-effective, dependable, and secure virtualization solution, 2011.
- [10] Siyuan Jing and Kun She. A generic approach to achieve optimal server consolidation by using existing servers in virtualized data center. *World Academy of Science, Engineering and Technology*, 75:1179–1185, 2011.
- [11] Andi Mann. Virtualization is not cloud... but cloud needs virtualization. <http://pleasediscuss.com/andimann/20091120/virtualization-is-not-cloud-%E2%80%A6-but-cloud-needs-virtualization/>, 2009.
- [12] Matthew L. Massie, Brent N. Chun, and David E. Culler. The ganglia distributed monitoring system: Design, implementation and experience. *Parallel Computing*, 30(7), July 2004.
- [13] National Institute of Standards and Technology U.S Department of Commerce. The nist definition of cloud computing (draft). http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf, jan 2011.
- [14] Department of Water Resources State of California. Department of water resources consolidates data center, reduces server footprint by two-thirds. <http://techblog.ca.gov/?p=2309>, aug 2011.

- [15] Gerald J. Popek and Robert P. Goldberg. Formal requirements for virtualizable third generation architectures. *Communications of the ACM*, 17(7), July 1974.
- [16] Qumranet. Kvm: Kernel-based virtualization driver - white paper. http://www.linuxinsight.com/files/kvm_whitepaper.pdf, 2006.
- [17] John Scott Robin and Cynthia E. Irvine. Analysis of the intel pentium's ability to support a secure virtual machine monitor. In *Proceedings of the 9th USENIX Security Symposium*. USENIX Association, 2000. http://www.usenix.org/events/sec00/full_papers/robin/robin.pdf.
- [18] James E. Smith and Ravi Nair. *Virtual Machines: versatile platforms for systems and processes*. Morgan Kaufmann, 2005.
- [19] Benjamin Speitkamp and Martin Bichler. A mathematical programming approach for server consolidation problems in virtualized data centers. *IEEE TRANSACTIONS ON SERVICES COMPUTING*, 3(4):266–278, October - December 2010.
- [20] Mueen Uddin and Azizah Abdul Rahman. Virtualization implementation model for cost effective & efficient data centers. *IJACSA*, 2(1):69–74, January 2011.
- [21] William Vambenepe. Look ma, no hypervisor! <http://stage.vambenepe.com/archives/976>, 2009.
- [22] VMware. Esx server 2 - security white paper. http://www.vmware.com/pdf/esx2_security.pdf.
- [23] Virtual Square Wiki. Virtual square networking. http://wiki.virtualsquare.org/wiki/index.php/Introduction#Virtual_Square_Networking, 2011.
- [24] Wikipedia. Physical to virtual. <http://en.wikipedia.org/wiki/Physical-to-Virtual>.

- [25] Timothy Wood, Ludmila Cherkasova, Kivanc Ozonat, and Prashant Shenoy. Profiling and modeling resource usage of virtualized applications. In *Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware*, pages 366–387, 2008. <http://portal.acm.org/citation.cfm?id=1496973>.

Ringraziamenti

Ringrazio la mia famiglia, che mi ha sempre aiutato e spinto a raggiungere questo traguardo; un grazie alla mia ragazza, che è riuscita a sopportarmi ed incoraggiarmi giorno dopo giorno, e a tutti gli amici, vecchi e nuovi, che mi sono stati vicini e che hanno reso questi anni di studio molto più piacevoli. Un ringraziamento particolare va infine al prof. Davoli, la cui disponibilità mi ha permesso di concludere questo lavoro nel poco tempo disponibile.