

Solucions per a la virtualització de servidors

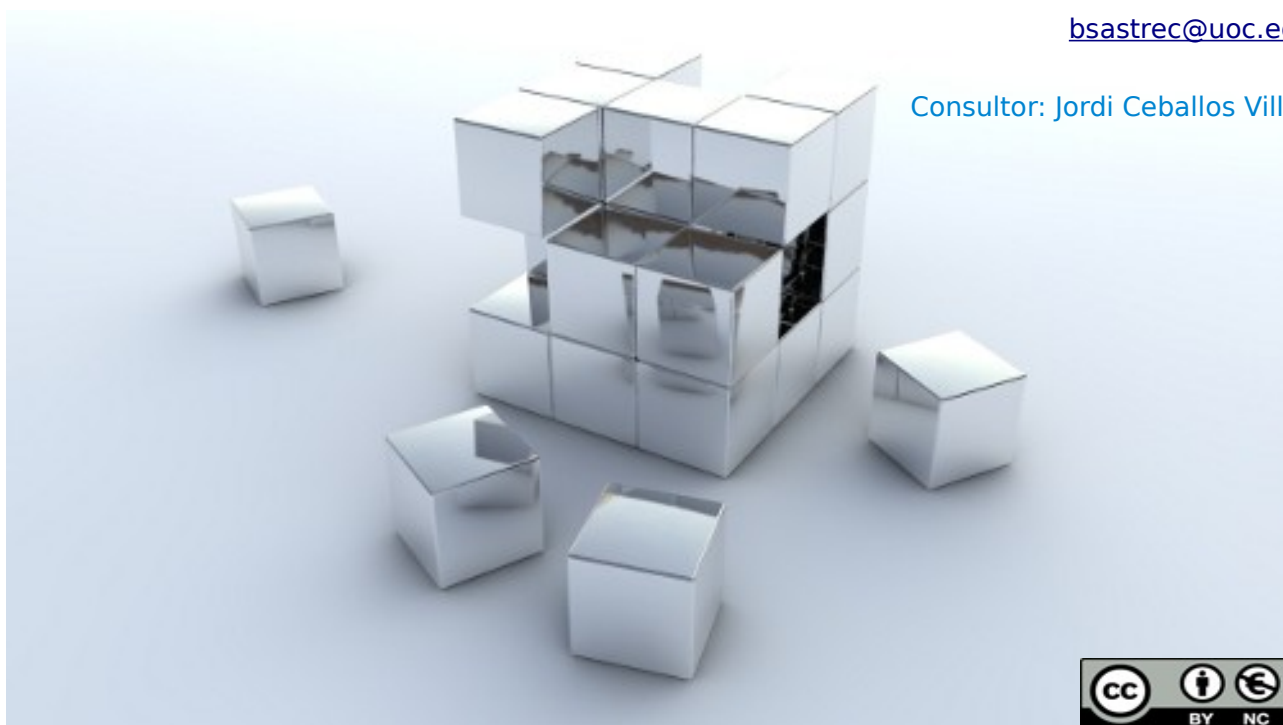
Projecte final de carrera Àrea Xarxes

Bartomeu F. Sastre Cabanellas

Enginyeria Informàtica
Juny 2012

bsastrec@uoc.edu

Consultor: Jordi Ceballos Villac



*A tots els companys que s'han anat creuant i acompanyat durant aquest viatge,
a tots els meus amics que tant m'han recolzat,
als consultors i tutors que amb el seu temps, coneixement i paciència m'han resolt els dubtes.*

i sobretot

Als meus pares, Josep i Catalina, que sempre m'han donat tot el seu amor.

Índex

1. Introducció.....	5
1.1. Descripció del projecte.....	5
1.3. Planificació.....	6
2. Virtualització.....	7
2.1. Què és la virtualització?.....	7
2.2. Història de la virtualització.....	8
2.3. Problemàtica de sistemes distribuïts.....	8
2.3.1. <i>Baixa utilització del sistema</i>	8
2.3.2. <i>Costos associats a la infraestructura</i>	8
2.3.3. <i>Augment dels costos de gestió</i>	8
2.3.4. <i>Protecció davant de desastre (failover)</i>	8
2.3.5. <i>El punt final del sistema</i>	9
2.4. Nova era en la virtualització.....	9
3. Avantatges dels sistemes virtualitzats.....	9
3.1. Aïllament.....	9
3.2. Seguretat.....	10
3.3. Flexibilitat.....	10
3.4. Agilitat.....	10
3.5. Recuperació.....	10
3.6. Reducció de costos.....	10
3.7. Exemple gràfic.....	11
4. Tipus de virtualització.....	14
4.1. Emulació o simulació.....	14
4.2. Virtualització nativa i virtualització completa.....	14
4.3. Virtualització parcial (hi incloem l'anomenat address space virtualization).....	15
4.4. Paravirtualització.....	15
4.5. Virtualització en l'àmbit del sistema operatiu.....	16
4.6. Virtualització d'aplicacions.....	16
4.7. Taula de programaris de virtualització a avaluar.....	16
5. Programari de virtualització.....	17
5.1. Solucions actuals i emergents.....	17
5.2. VMware ESXi 5.0.....	18
5.2.1. <i>Característiques principals</i>	19
5.3. Citrix Xen Server 6.0.....	21
5.4. Microsoft Hyper-V.....	23
5.5. Proxmox VE 2.0.....	24
5.5.1. <i>Kernel-based Virtual Machine (KVM)</i>	25
5.5.2. <i>QEMU</i>	25
5.5.3. <i>OpenVZ</i>	26
5.6. RHEV 3.0 (Red Hat Enterprise Virtualization).....	26
6. Interfícies de treball	28
6.1. VMware ESX 5.0.....	29
6.1.1. <i>Servidor</i>	29
6.1.2. <i>Client</i>	31
6.2. Microsoft Hyper-V 2008 R2.....	33
6.3. Proxmox 2.0.....	34
6.3.1. <i>Servidor</i>	34
6.3.2. <i>Client</i>	36
6.4. Red Hat Enterprise Linux-RHEL.....	37

6.4.1. Servidor.....	38
6.5. Citrix Xen Server 6.0.....	39
6.5.1. Servidor.....	39
6.5.2. Client.....	40
7 Benchmarks.....	42
7.1. Introducció.....	42
7.2. Maquinari/programari sistema operatiu bàsic.....	42
7.3. Disseny del sistema.....	44
7.4. Eines de benchmarks emprades.....	45
7.4.1. Phoromix Test Suite.....	45
7.4.2. Instal·lació suite.....	45
7.4.3. Scripts PTS.....	46
7.5. CPU - Processador.....	48
7.5.1. Compress-7-Zip.....	48
7.5.2. Compressió gZip.....	50
7.5.3. Mitjana dels càlculs.....	52
7.5.4. Resultat obtinguts.....	53
7.6. RAM+memòria.....	54
7.6.1. Stream.....	54
7.6.2. Gràfiques-resultats.....	55
7.6.3. Resultat obtinguts.....	56
7.7. Discs I/O.....	57
7.7.1. FS-Mark.....	57
7.7.2. Gràfiques de rendiment.....	58
7.7.3. Postmark.....	59
7.7.4. Gràfiques.....	60
7.7.5. Mitjana dels resultats.....	61
7.7.6. Resultats obtinguts.....	62
7.8. Sistema.....	63
7.8.1. PHPBench.....	63
7.8.2. Gràfiques.....	64
7.8.3. Apache.....	65
7.8.4. Gràfiques.....	66
7.8.5. Mitjana.....	67
7.8.6. Resultats obtinguts.....	68
7.9. Xarxa.....	69
7.9.1. Script test xarxa.....	69
7.9.2. Generació de fixers – crear.sh.....	71
7.9.3. Còpies.....	71
7.9.4. Problemàtica.....	72
7.9.5. Gràfiques.....	75
7.9.6. Resultats obtinguts.....	80
7.10. Resum estadístic general.....	81
7.10.1. Resultat finals. Conclusions.....	82
8 Conclusions.....	84
9 Línies de Millora.....	85
9.1 Proxmox, proves de HA.....	85
9.2 Realitzar més proves de CPU.....	85
9.3 Test de disc sobre altres tecnologies.....	85
9.4 Unitat de dades no virtualitzada.....	85
9.5 Driver Hyper-V.....	85
10 Bibliografia i Referències consultades.....	86
10.1 Història de la virtualització.....	86
10.2 Tipus de virtualització.....	86
10.3 Programari de Virtualització.....	86
10.4 Eines de Benchmark.....	86

1. Introducció

En el món de la informàtica tot canvi és continu, el que fins avui dia era una eina o opció inamovible de cop i volta es transforma en una tecnologia o eina obsoleta. El nostre és un món fascinant de canvis i innovacions constants.

En l'àmbit de les xarxes i sistemes, tal volta semblava que ho havíem vist tot fins que va arribar aquest terme que qui l'empra ho veu molt senzill, però que com a regla general se'ns fa un tant difícil d'explicar als profans la virtualització de servidors. És aquesta tecnologia, la virtualització, la que està fomentant un present i un futur en el món de la gestió de la informàtica. Tracta d'una nova filosofia a l'hora de gestionar els serveis que oferim als usuaris, ja que no parlarem de la gestió de servidors físics com fins ara havíem fet, sinó que parlarem de gestionar serveis, a causa dels avantatges que ens portarà poder virtualitzar els servidors, que fins ara eren màquines físiques lligades a uns serveis donats.

Gràcies a la virtualització, aquest lligam de màquina física i servei es desvincula totalment, i dóna peu a una forma molt més flexible i impensable fins fa poc per al món de la gestió i administració de servidors.

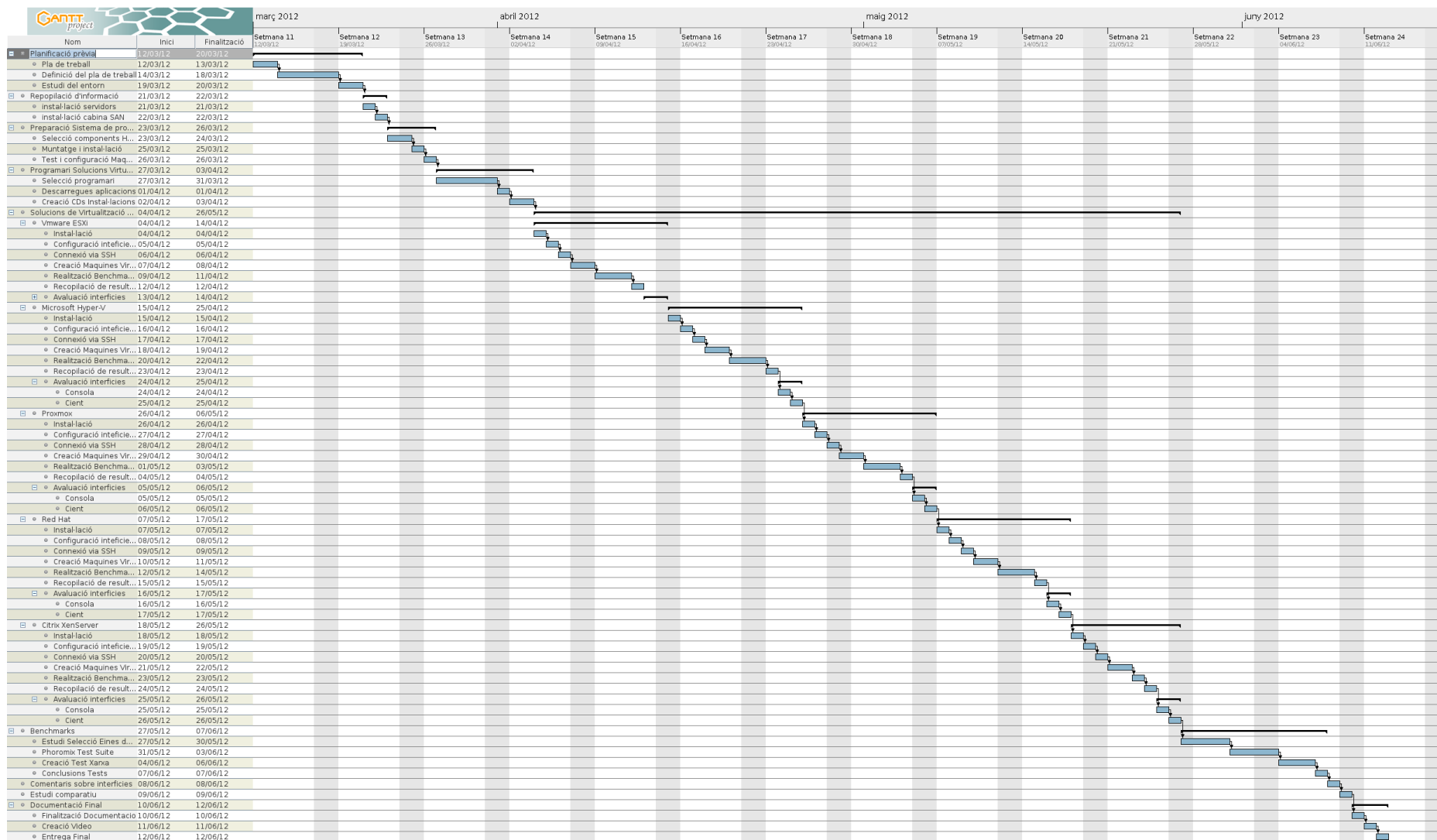
1.1. Descripció del projecte

El projecte consistirà en la descripció i aprofundiment en el món de la virtualització: què és?, per a què serveix? Com s'empra?

Parlarem de les solucions virtuals més emprades avui dia en el món laboral i empresarial, els farem una sèrie de tests i veurem el seu rendiment per saber fins a on poden arribar. Amb les dades obtingudes podrem veure en quina situació es troba aquest nou paradigma, al mateix temps que intentarem extreure conclusions sobre els resultats.

Finalment, comentarem les peculiaritats que més ens han sorprès de cada producte, tan com el seu funcionament, instal·lació i interfície de treball.

1.3. Planificació



2. Virtualització

2.1. Què és la virtualització?

La virtualització és, segons una definició estricta, l'abstracció dels recursos d'un ordinador anomenat *Hypervisor Virtual Machine Monitor* (VMM), el qual crea una capa d'abstracció entre el programari de la màquina física i el sistema operatiu de la màquina o màquines virtuals.

Aquesta capa de programari (VMM) és l'encarregada de gestionar els recursos de l'ordinador, CPU, memòria, emmagatzemament i xarxa, de tal forma que podrà repartir dinàmicament els recursos esmentats entre les diferents màquines virtuals contingudes a l'ordinador central.

Dit d'una forma menys abstracta, podem dir que es tracta que, mitjançant una capa de programari, que se situa sobre una màquina física, podem gestionar sobre ella un nombre de màquines virtuals, les quals comparteixen els recursos de la màquina física o *host*, sense adonar-se l'usuari d'aquesta condició i podent-les utilitzar sense notar cap diferència respecte a una màquina física.

La virtualització és una capa abstracta que desacobla el maquinari físic del sistema operatiu per oferir una major flexibilitat i utilització dels recursos de TI.

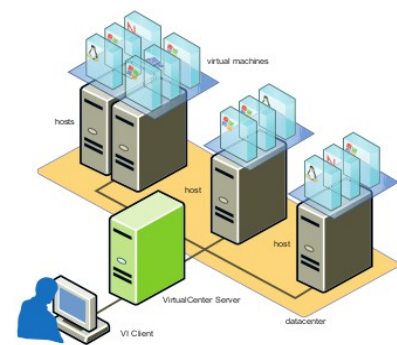


Illustration A: Esquema sobre virtualització

La virtualització permet que múltiples màquines virtuals amb sistemes operatius heterogenis es puguin executar individualment, encara que en la mateixa màquina. Cada màquina virtual té el seu propi maquinari virtual (per exemple, RAM, CPU, NIC, etc.), a través del qual es carreguen el sistema operatiu i les aplicacions. El sistema operatiu distingeix el maquinari com un conjunt normalitzat i consistent, independentment dels components físics que realment en formin part.

Les màquines virtuals s'encapsulen en arxius i permeten guardar, copiar i proporcionar una màquina virtual de manera ràpida. Es poden moure en segons quins sistemes sencers (aplicacions, sistemes operatius, BIOS i maquinari virtual completament configurats) d'un servidor a un altre amb consolidació contínua de treball i un manteniment sense temps d'inactivitat.

La virtualització es va introduir inicialment als anys 60 per permetre la divisió de grans unitats de maquinari *mainframe*, un recurs costós i escàs. Amb el temps, les minicomputadoras i PC van proporcionar una manera més eficient i assequible de distribuir el poder de processament, de manera que en els anys 80 la virtualització ja gairebé no es va utilitzar més.

En els anys 90, els investigadors van començar a veure com la virtualització podia solucionar alguns dels problemes relacionats amb la proliferació de maquinari menys costós, incloent la seva subtilització, creixents costos d'administració i vulnerabilitat.

Avui dia, la virtualització està a l'avantguarda, ajudant els negocis amb l'escalabilitat, seguretat i administració de les seves infraestructures globals de TI.

2.2. Història de la virtualització

Els inicis de la virtualització queden datats als laboratoris d'IBM al voltant dels anys 60. La idea inicial era la de particionar els grans ordinadors centrals (*mainframes*) que desenvolupava la companyia, en màquines independents dins el mateix *mainframe*.

Aquestes particions permetien als *mainframes*, com ara l'IBM 7044, poder realitzar operacions amb diverses aplicacions i processos a la vegada. El motiu d'haver invertit en la tecnologia esmentada va ser econòmic. Hem de tenir en compte que en aquella època els dits equips i els seus components eren molt cars i, com que es podien particionar, s'aprofitava de millor manera la màquina i la seva inversió.

No obstant el paradigma que hi havia als anys 60 de la centralització de la computació i de tenir els terminals del tipus client lleuger (*thin client*), ordinadors bàsics, desproveïts de disc dur (o sistema d'emmagatzemament similar) i sistema operatiu pròpiament instal·lat, es va deixar de seguir a les dècades dels 80 i 90 a causa de l'abaratiment dels costos del maquinari i la creació dels sistemes distribuïts; és a dir, el fet que no recaigués tota la tasca de procés sobre els ordinadors centrals i s'anàs repartint entre els clients o altres màquines servidores, basades en l'arquitectura x86, en lloc d'una sola i costosa d'una empresa en concret.

2.3. Problemàtica de sistemes distribuïts

Encara que en un principi el baix cost dels equips basats en arquitectures x86 semblava un avantatge significatiu, a la llarga varen sorgir una sèrie de problemàtiques que feren que aquesta solució no fos la més idònia per al manteniment i gestió del sistema.

2.3.1. Baixa utilització del sistema

Atès el baix cost, se sol disposar d'un servidor per servei, la qual cosa fa que els servidors realment estiguin infrautilitzats, ja que segons els estudis realitzats només s'empra un 10 % o un 15 % de la seva capacitat.

2.3.2. Costos associats a la infraestructura

A mesura que va creixent el sistema, s'augmenta el nombre de servidors físics, fet que du associat un increment de la quantitat d'espai físic a emprar, una major despesa econòmica, adequació de l'espai a on estan allotjats, inversió major en sistemes de refrigeració i tot allò que sigui necessari per a la millora del CPD (centre de procés de dades).

2.3.3. Augment dels costos de gestió

Atès que els sistemes van creixent, més complex es fa el manteniment de tot el conjunt, no només pel que fa al servidor, sinó també a les infraestructures per interconnectar-los i tota l'electrònica necessària.

2.3.4. Protecció davant de desastre (*failover*)

Inversió en la reducció del temps d'aturada, s'haurà d'apostar per sistemes de SAI, alternadors,

cosa que també redundarà físicament els servidors; és a dir, crear clústers de servidors per minimitzar una possible aturada física, amb la qual cosa tenim un servidor duplicat que en principi no fa res, i possiblement al llarg de la seva vida mai s'hagi de necessitar el seu ús.

2.3.5. El punt final del sistema

Els escriptoris dels usuaris, el manteniment dels seus computadors es converteix en una altra tasca de manteniment repetitiva i generalment allò que causa el 90 % de les cridades al departament de suport tècnic. Per altra banda, s'ha de tenir en compte la qüestió de la seguretat associada als escriptoris: a major nombre de terminals, es fa molt més complex de mantenir.

2.4. Nova era en la virtualització

Arran de totes les problemàtiques abans comentades, es va començar a pensar de nou en la centralització dels equips en lloc de la distribució que havia governat en la dècada 80-90. A tal efecte, es va produir l'entrada de l'empresa VMware, la qual va tornar a fer ressorgir la tècnica de la virtualització que havia començat IBM. La diferència és que es va implementar sobre la plataforma x86, que ja havia arrelat fortament en les empreses i institucions, i el cost de la qual era relativament econòmic.

Vista aquesta implantació, s'observaren una gran quantitat d'avantatges que rectificaven els problemes de què parlàvem als punts anteriors (utilització del sistema, costos de la infraestructura, costos de gestió, protecció – *failover*, escriptoris usuari).

3. Avantatges dels sistemes virtualitzats

Quant a l'aprofitament del programari, com sabem als CPD tenim un gran nombre de servidors dedicats a tasques úniques, sovint infrautilitzats. Podria ser el cas dels servidors redundants, que són còpies d'altres i només funcionaran quan el principal caigui. Pot ser que no es produeixi mai, però mentrestant hem de realitzar el manteniment metòdic d'aquesta segona màquina.

Si es du a terme una virtualització i agrupam en el nombre necessari servidors físics, molts dels servidors abans infrautilitzats quedaran plenament utilitzats, s'estalviarà en energia, espai, capacitat de refrigeració i administració.

Els avantatges, els podem veure en els següents punts.

3.1. Aïllament

Les màquines virtuals són independents les unes de les altres gràcies a la capa de programari *Hypervisor*.

L'aïllament ens assegura que la caiguda d'un servidor virtual no afectarà la resta de servidors,

que continuaran realitzant les seves tasques.

3.2. Seguretat

Cada màquina virtual té un alt nivell d'accés del tipus *root* independent a la resta de les màquines; així doncs, un atac de seguretat a una màquina virtual només afectarà aquesta. El present punt està lligat amb l'anterior.

3.3. Flexibilitat

Tenim tanta flexibilitat com limitacions físiques tengui el nostre servidor físic (o clúster) a l'hora de crear nous servidors virtuals. És a dir, podem crear un servidor virtual amb la RAM, CPU, disc que vulguem sense haver d'adquirir més equips ni recursos addicionals.

També podem tenir tot tipus de sistemes operatius executant-se, encara que depenent del tipus de virtualització també anirà lligat a l'arquitectura del servidor físic.

3.4. Agilitat

A l'hora de crear noves màquines virtuals, ens trobarem amb un procés francament senzill, la creació de màquines virtuals a nivell arquitectura, es pot parlar d'escassos minuts i fins i tot segons si es tracta d'una configuració estàndard.

Hi ha solucions que ens permeten ja tenir màquines semi instal·lades, plantilles, de tal forma que fins i tot el fet monòton d'instal·lar el sistema operatiu ja només s'haurà de realitzar un cop per versió, o tenir ja servidors amb determinades configuracions de programari.

Portabilitat/seguretat

La simplicitat de la configuració d'una màquina virtual resideix que està emmagatzemada en fitxers de text pla, a part que el que seria el disc dur d'aquesta també és un altre fitxer (tan gran com siguin els discos); per tant, copiar aquests fitxers a un altre entorn virtualitzat ja ens permetrà tenir una còpia exacta del servidor. També podrem utilitzar-la per clonar màquines i poder experimentar sobre elles en lloc de les màquines en producció.

3.5. Recuperació

Està associada al punt de la portabilitat, pel fet de tenir una màquina exportada i amb la facilitat que du implícit tornar a posar-la en marxa. Ens dona una seguretat molt gran si es produeixen problemes greus als servidors físics i hem d'aixecar de bell nou el sistema en poc temps.

3.6. Reducció de costos

Es minimitza l'espai físic que necessitam al CPD per al maquinari físic. Aleshores, com que hi ha menys màquines físiques, també minva el consum elèctric per dues vies: reducció de

consum energètic i, com que hi ha menys màquines, el consum de refrigeració també disminueix.

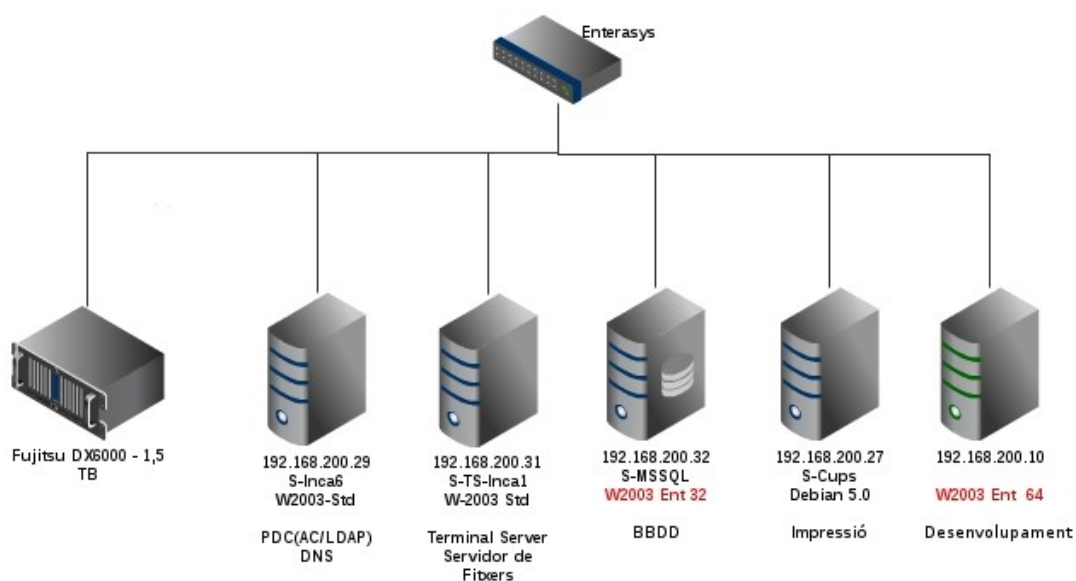
3.7. Exemple gràfic

Per veure d'una manera un tant gràfica el canvi que implica la virtualització de servidors, ens basarem en els següents esquemes.

En el primer esquema podem veure la instal·lació tradicional d'un centre de càlcul, el qual consta d'una sèrie de servidors: servidor de domini, *terminal server*, servidor de base de dades, impressió i un servidor de desenvolupament.

A més, per reduir el tràfic LAN, necessitarem de l'electrònica necessària per connectar-los entre ells.

Podem veure que necessitarem l'espai físic per a aquests cinc servidors, el més normal és que es faci instal·lant-los a dins un *rack* (bastidor) per ocupar el menor espai; en cas que no parlem de servidors muntats en bastidors, l'espai necessari seria encara molt major.



Il·lustració B: Entorn tradicional sense rack

En la següent imatge podem veure com quedaria l'esquema d'abans, amb els servidors en bastidors. Es pot apreciar una millora de l'espai substancial. Però, els desavantatges de seguir emprant una instal·lació tradicional se segueixen mantenint.

Cal tenir en compte la complexitat del cablejat necessari per interconnectar tot el sistema, encara que *a priori* no ho pugui semblar. Aquest fet implica disposar del cablejat necessari per a tal finalitat; per tant, necessitariem com a mínim 2 cables de xarxa RJ-45 per servidor connectats al commutador, a més dels cables necessaris per a la seva gestió, com ara el teclat, el ratolí i el monitor, sense oblidar-nos de dos cables de corrent per servidor. Així, no s'oblida la qüestió del cablejat, que val tenir present a causa del seu volum.

En aquesta imatge podem veure el cablejat de la part del darrere d'un rack en què s'aprecien els distints tipus de cablejat abans esmentats, corrent, dades i perifèrics.



Illustration D: Detall cablejat de servidors

Com que hi ha més servidors, si no es fa una planificació detallada i acurada de la part del darrere dels racks, la problemàtica del cablejat pot arribar a ser bastant delicada i de costosa resolució.

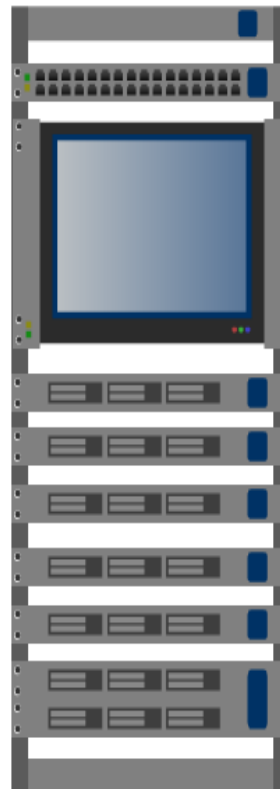


Illustration C: Instal·lació tradicional dins



Illustration E: Cablejat de rack sense sanejar

Al fet d'arreglar un *rack*, és a dir, col·locar correctament i de forma ordenada el cablejat se li diu *sanejar*. Hi ha empreses de telecomunicacions especialitzades en aquesta tasca. No és una feina extremadament delicada, però sí bastant laboriosa.



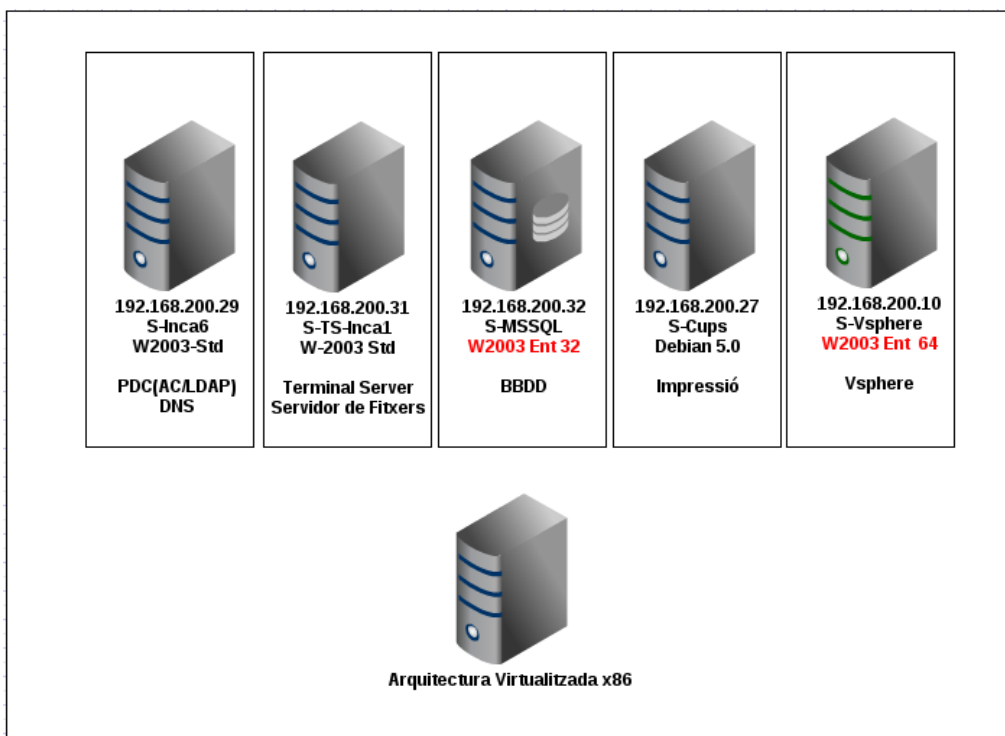
Illustration F: Rack servidor amb solució virtualització

Per contra, el fet de virtualitzar els servidors ens permet simplificar enormement el tema de cablejat i el tema de l'espai.

A nivell físic, podríem aprofitar el *rack* que havíem vist abans amb un sol servidor; a dins ell, hi tendríem virtualitzats tots els servidors i cada un d'ells realitzaria una tasca determinada.

Dit d'una altra forma, la disposició física dels servidors es transforma en una disposició lògica, mitjançant la virtualització.

Com veiem a l'esquema següent, l'esquema lògic de funcionament és semblant a l'esquema físic del que seria una instal·lació estàndard. Veim com a dins a un sol servidor tenim els cinc servidors que abans teníem en disposició física.

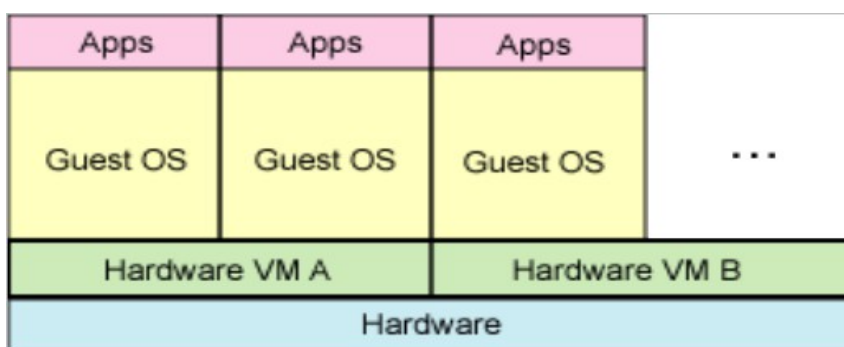


Il·lustració G: Entorn virtualitzat

4. Tipus de virtualització

4.1. Emulació o simulació

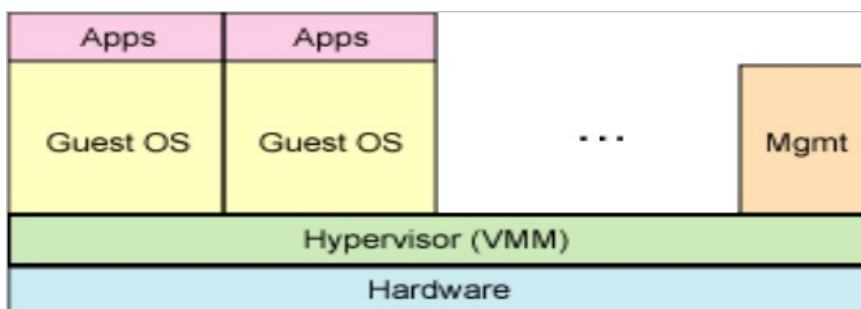
La màquina virtual simula un maquinari complet, admetent un sistema operatiu *guest* sense modificar per una CPU completament diferent. Aquest enfocament va ser molt utilitzat per permetre la creació de programari per a nous processadors abans que estiguessin físicament disponibles. Per exemple Bochs, PearPC, Qemu sense acceleració, i l'emulador Hercules. L'emulació és posada en pràctica utilitzant una varietat de tècniques, des d'autòmats d'estats (*state machines*) fins a l'ús de la recopilació dinàmica en una completa plataforma virtual.



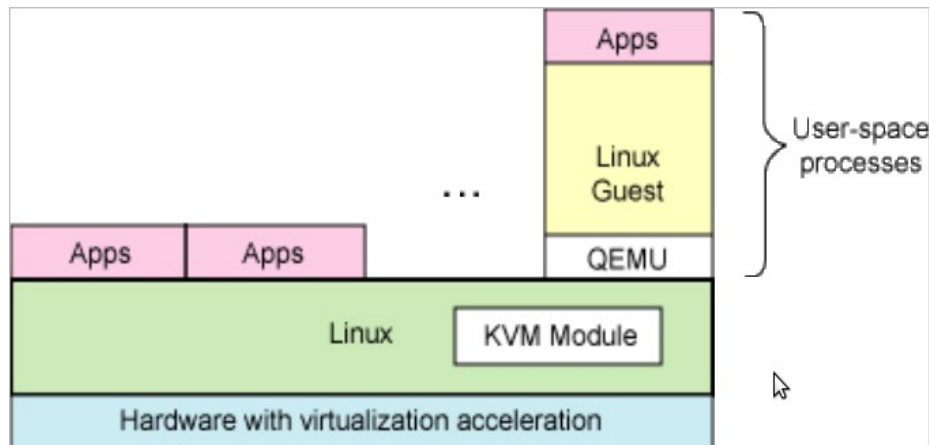
Il·lustració H: Arquitectura Qemu

4.2. Virtualització nativa i virtualització completa

En aquest cas, la màquina virtual simula un maquinari suficient per permetre un sistema operatiu *guest* sense modificar (un de dissenyat per a la mateixa CPU) per córrer de forma aïllada. Típicament, moltes instàncies poden córrer al mateix temps. Aquest enfocament va ser el pioner el 1966 amb CP-40 i CP [-67] / CMS, predecessors de la família de màquines virtuals d'IBM. Alguns exemples: VMware Workstation, VMware Server, Parallels Desktop, Virtual Iron, Adeos, Mac-on-Linux, Win4BSD, Win4Lin Pro yz / VM.



Il·lustració I: Arquitectura VMware ESX



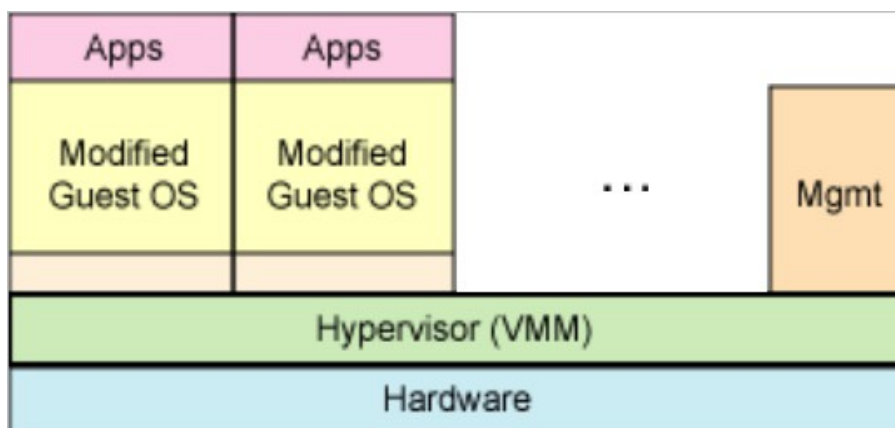
Il·lustració J: Arquitectura KVM

4.3. Virtualització parcial (hi inclouem l'anomenat *address space virtualization*)

Veurem com aquí la màquina virtual simula múltiples instàncies de molt (però no de tot) de l'entorn subjacent del maquinari, particularment *address spaces*. Aquest entorn admet compartir recursos i aïllar processos, però no permet instàncies separades de sistemes operatius *guest*. Encara que no és vista com dins de la categoria de màquina virtual, històricament aquest va ser un important acostament i va ser utilitzat en sistemes com CTSS, l'experimental IBM M44/44X, i es podria dir que en d'altres com OS/VS1, OS/VS2 i MVS.

4.4. Paravirtualització

En la paravirtualització, la màquina virtual no necessàriament simula un maquinari, en canvi ofereix un API especial que només es pot utilitzar mitjançant la modificació del sistema operatiu *guest*. La crida del sistema a l'hipervisor té el nom d'*hypercall* a Sen i Parallels Workstation; està implementada via el maquinari *instruction DIAG (diagnose)* al CMS VM en el cas d'IBM (aquest va ser l'origen del terme *hypervisor*). Exemple: VMware ESX Server, Win4Lin 9x i z / VM.



Il·lustració K: Arquitectura Paravirtualitzada

4.5. Virtualització en l'àmbit del sistema operatiu

Virtualitzar un servidor físic en l'àmbit del sistema operatiu permet a múltiples servidors virtuals aïllats i assegurances córrer en un sol servidor físic. L'entorn del sistema operatiu *guest* comparteix el mateix sistema operatiu que el del sistema *host* (el mateix nucli del sistema operatiu és usat per implementar l'entorn del *guest*). Les aplicacions que corren en un entorn *guest* donat ho veuen com un sistema autònom. Exemples: Linux-vserver, Virtuozzo, OpenVZ, Solaris Containers i FreeBSD Jails.

4.6. Virtualització d'aplicacions

Consisteix en el fet de córrer a un escriptori o una aplicació de *server* localment, usant els recursos locals, en una màquina virtual apropiada. Aquest fet contrasta amb córrer l'aplicació com un programari local convencional (programaris que van ser "instal·lats" en el sistema). Semblants aplicacions virtuals corren en un petit entorn virtual que contenen els components necessaris per executar entrades de registres, arxius, entorns variables, elements d'ús d'interfícies i objectes globals. L'entorn virtual esmentat actua com una capa entre l'aplicació i el sistema operatiu, i elimina els conflictes entre aplicacions i entre les aplicacions i el sistema operatiu. Els exemples inclouen el Java Virtual Machine de Sun, Softricity, Thinstall, Altiris i Trigenice (aquesta metodologia de virtualització és clarament diferent a les anteriors; només una petita línia divisòria els separa d'entorns de màquines virtuals com Smalltalk, FORTH, Tel, P – code).

4.7. Taula de programaris de virtualització a avaluar

En aquesta taula exposam els programaris de virtualització més reconeguts i estesos actualment en les implantacions i mercat.

Nom	Desenvolupador	SSOO <i>host</i> /requisit	Llicència	Tipus de virtualització
Hyper-V	Microsoft	Windows Server 2008	Propietària, inclosa a dins el Windows 2008 Hypervisor Server	
Proxmox KVM	Qumranet [4]	Solaris, Windows, Linux, Mac OS X	Hiku, GPL version 2	Virtualització nativa Paravirtualization
Proxmox QEMU	Fabrice Bellard helped by other developers	Windows, Mac OS X	Hiku, Linux, GPL/LGPL	Emulador de processador
Proxmox QEMU w/ <i>kqemu</i> module	Fabrice Bellard	Windows, Mac OS X	Hiku, Linux, GPL/LGPL	
Red Hat Enterprise Virtualization	Red Hat	Red Hat Linux	Propietària gratuïta	Basat sobre el KVM
VMware Server	ESXi VMware	Linux – Propietari VMware	Propietària gratuïta	Bare-metal embedded hypervisors
XenServer	Citrix Systems	Linux – Propietari Citrix	Propietària gratuïta	Paravirtualization

5. Programari de virtualització

5.1. Solucions actuals i emergents

En l'ampli marc actual de la virtualització tenim una sèrie de noms sobre solucions a tenir en compte.

Hi trobarem marques comercials que són tota una institució en aquest camp, com és el cas de VMware, que hi ha estat des de sempre una empresa que ha sobresortit. Però, atesa l'emergència d'aquestes tecnologies i la gran demanda que ha anat sorgint al voltant d'ella, moltes empreses han desplegat i presentat les seves pròpies solucions. Algunes d'elles ja fa temps que tenien productes en el mercat, però ara hi retornen amb més força, com el cas de Microsoft amb el seu Hyper-V. D'altres, per no perdre temps i poder entrar amb força al mercat, es decanten per adquirir solucions ja existents en el món del programari lliure, comprar-les i adaptar-les per a la seva comercialització, com és el cas de Citrix XenServer quan adquirí el XenSource, que era de la Universitat de Cambridge.

Altres empreses segueixen altres camins de la virtualització des del món del programari lliure, com és el cas de Red Hat. Aquesta, com a empresa que desenvolupa programari lliure i ofereix els seus serveis a les empreses, ha apostat per seguir emprant la seva plataforma i sobre ella instal·lar-hi solucions lliures com ara el KVM i el QEMU per donar servei a la virtualització.

Una altra empresa que tecnològicament és molt semblant a Red Hat és Proxmox Server Solutions GmbH, que ens presenta una solució anomenada Proxmox Virtual Environment (Proxmox VE) basada sobre el sistema operatiu Debian i, igual que Red Hat, com a eines de virtualització emprava KVM i QEMU.

Aquesta solució té com a particularitat que és totalment lliure i gratuïta, a més de tenir una sèrie de funcionalitats que les altres solucions no aporten en les seves versions bàsiques, com ara l'alta disponibilitat (HA).

5.2. VMware ESXi 5.0

De la mà de l'empresa VMware, disposam de la solució anomenada ESXi, també anomenada VMware vSphere Hypervisor, actualment és la versió 5.0 la que tenim disponible. L'ESXi és família de la versió anomenada ESX (sense la 'i'), la diferència és que l'ESXi és gratuïta i és sobre la que realitzarem les proves.

La família ESX està basada sobre el *bare-metal hypervisors*, és a dir, són els hipervisors de VMware que treballen directament sobre el maquinari sense haver de requerir d'una capa addicional de sistema operatiu.

El seu sistema es troba al nivell inferior de la capa de virtualització, l'hipervisor, però disposa d'eines i serveis autònoms i independents, que es poden executar remotament.

Està compost per un sistema operatiu autònom, el qual ens proporciona l'entorn de gestió, administració i execució del programari hipervisor, i tota la interacció entre aquest i les màquines virtuals.

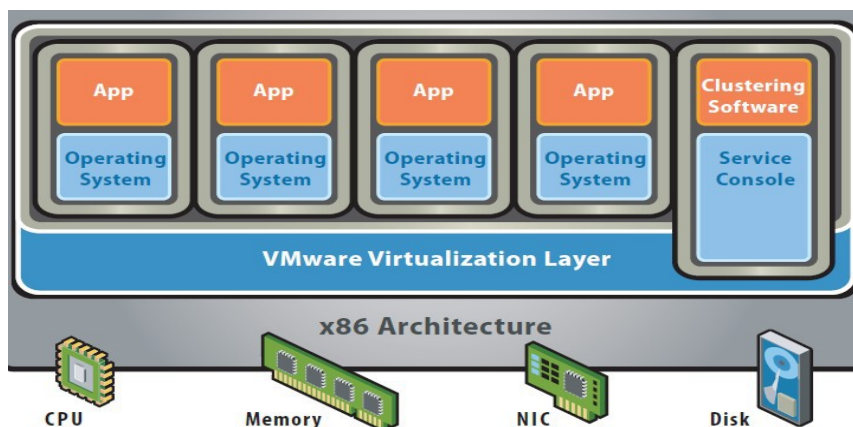
Com hem dit es tracta d'una plataforma del tipus hipervisor, no s'executa sobre un sistema operatiu específic, sinó que està a dins el seu mateix nucli, el qual està basat sobre un sistema Linux, en concret el Red Hat Enterprise Linux (RHEL) adaptat a les particularitats de VMware. L'avantatge d'aquest fet és que, si ja disposam de coneixement en el sistema operatiu GNU-Linux, tendrem major facilitat a l'hora d'emprar la consola i les instruccions específiques de VMware (CLI) per manipular les màquines.

Les versions superiors de l'ESXi, les ESX, disposen d'una sèrie de característiques avançades, a majors funcionalitats, major preu de les llicències a pagar i major el manteniment anual d'aquestes; per tant, cal tenir ben clara la versió a adquirir.

L'arrencada de la màquina amfitriona es produeix a través de l'execució d'un nucli Linux, el qual proporciona serveis de consola i maquinari en l'àmbit de l'anell 04 en col·laboració amb l'hipervisor funcionant en mode "supervisor". A partir de la versió vSphere (versió 4.0), l'hipervisor aplica els teoremes de la paravirtualització i substitueix el nucli Linux per les seves pròpies interfícies creant un nivell d'anell -1 i passant a executar l'entorn operatiu com una màquina virtual.⁵

El VMkernel està programat i configurat seguint l'arquitectura de *microkernel* 6 i té les següents interfícies amb l'exterior:

- Programari
- Sistema convidat (*guest*)
- Consola de servei (Console OS, Service Console)

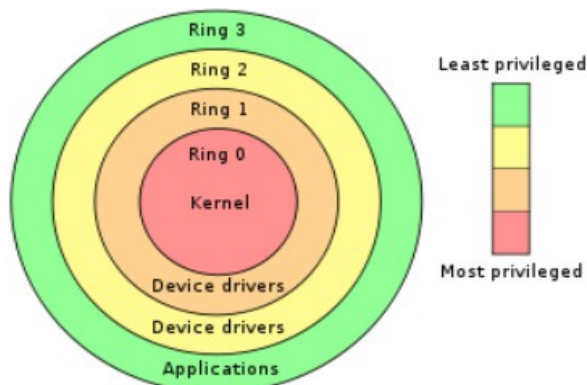


Bare-Metal (Hypervisor) Architecture

Il·lustració L: Arquitectura del vmware ESX

Els anells abans esmentats són mecanismes de protecció de dades i funcionalitat de fallades, i comportament maliciosos.

Els sistemes operatius restringeixen l'accés de les aplicacions als recursos mitjançant aquests nivells de seguretat, els anells.



Il·lustració M: Esquema lògic dels anells

5.21. Característiques principals

A part de les funcionalitats bàsiques per a aquest tipus d'eines, com la gestió de màquines, creació, modificació, l'ESXi 5.0 disposa d'aquestes:

- **Generador d'imatge.** Un nou conjunt d'utilitats des de línia de comandament (CLI) que

permet als administradors crear imatges personalitzades. Aquest generador d'imatges pot ser usat per crear-ne d'adequades per a diferents tipus d'implementació, com ara instal·lacions via ISO o basades via PXE i desplegament automàtic.

- **VMware vShield.** Es tracta d'un complement que ens permet assegurar la infraestructura de virtualització i disposa d'aquestes característiques:
 - Control d'accés basat en rols.
 - Enrunament estàtic.
 - Zones IP *multi-tenant*.
 - Opcions de seguretat de dades que comproven el flux de dades que es mouen per l'entorn de virtualització.
- **Firewall ESXi.** La interfície d'administració està protegida per un tallafocs orientat als serveis que es pot configurar utilitzant el client vSphere o en la línia de comandes. Aquest nou sistema elimina la necessitat d'emprar l'eina Optables.
- Per als *hosts* remots, podeu especificar les adreces IP o rang d'adreces IP que tenen permís per accedir a cada servei.
- **Suport de SNMP.** ESXi 5.0 amplia el suport per SNMP v.2, amb un control complet per a tot el maquinari al *host*.
- **Syslog seguretat.** Tots els missatges de registre estan generats per Syslog i els missatges (*logs*) es poden enregistrar des de qualsevol servidor local o remot. Per accedir als logs del registre, s'hi pot connectar de forma remota segura emprant el protocol Secure Socket Layer (SSL) o les connexions TCP amb vSphere 5.0. La configuració de registre de *logs* es pot realitzar també utilitzant ESXCLI i també a través del client vSphere.
- **Gestió central de la imatge de host i la configuració a través d'implementació automàtica.** La combinació de les característiques dels perfils, generador d'imatges i PXE, la implementació de VMware vSphere automàtic simplifica la tasca de la gestió d'instal·lació de l'ESXi, així com l'actualització de centenars de màquines. Les imatges ESXi estan emmagatzemats de forma centralitzada a la biblioteca Auto Deploy.
- **Unificat CLI framework.** Un *framework* ESXCLI ampliat i millorat ofereix un ric conjunt d'ordres consistents i extensibles, incloent nous comandaments per facilitar l'acollida en la solució de problemes i manteniment. El *framework* permet la consistència d'autenticació, funcions i auditoria, i utilitza els mateixos mètodes que els *framework* de gestió com ara vCenter Server i PowerCLI.
- **Maquinari virtual d'ESXi.** Les màquines a virtualitzar poden constar del següent maquinari virtual:
 - Màquines virtuals de fins a 32 CPU virtuals, cosa que els permet executar grans càrregues de treball intensives de la CPU a la plataforma VMware ESXi.
 - S'hi pot assignar fins a 1 TB de memòria RAM.
 - Compatible amb gràfics en 3D per executar Aero de Windows i les aplicacions bàsiques de 3D en màquines virtuals.
 - Els dispositius USB 3.0 connectats a l'equip client que executa el client web o el client vSphere poden ser connectats a una màquina virtual i es pot accedir al seu interior. Cal dir que el *host* com a tal no els reconeix.
 - UEFI BIOS virtual. Les màquines virtuals que s'executen en ESXi 5.0 poden arrencar i utilitzar la interfície unificada Extended Firmware (UEFI).

Altres importants capacitats disponibles amb ESXi des de la versió 4.1:

- **Integració AD.** Capacitat per configurar el *host* per unir-se a un domini d'Active Directory, i qualsevol usuari que intenta accedir a la màquina automàticament s'autentica amb el

directori d'usuari centralitzat.

- **Instal·lació via scripts.** Capacitat per realitzar la instal·lació automàtica del programari ESXi en el disc local d'un servidor. Diversos mètodes d'implementació són suportats, inclosa l'arrencada de l'instal·lador d'ESXi d'un CD o a través de PXE i l'accés a l'arxiu de configuració a la xarxa mitjançant una varietat de protocols, com HTTP segur. A l'arxiu de configuració també podem donar les següents seqüències d'ordres per ser executades durant la instal·lació:
 - Preinstal·lació
 - Postinstal·lació
 - En primer lloc, d'arrencada

Aquests scripts s'executen localment al *host* ESXi, que pot realitzar diverses tasques com la configuració de xarxes virtuals del *host* i unir-la a vCenter Server.

- **Arrencada des de suport SAN per ESXi.** Aquest suport inclou SANT Fibre Channel, així com iSCSI i FCoE per als adaptadors d'emmagatzematge d'alguns que s'han classificat per aquesta capacitat.

5.3. Citrix Xen Server 6.0

Citrix Systems és una corporació multinacional fundada al 1989, que subministra tecnologies de virtualització, connexió en xarxa i programari com a serveis (ASSA) i informàtica en el *cloud*. És també molt lligat el seu nom al servidor d'aplicacions i el seu protocol ICA.

A l'any 2007, Citrix Systems, veient com el futur passava per poder disposar d'una aplicació de virtualització per tal de poder oferir-la als seus clients i poder entrar en el mercat competitiu d'aquesta especialitat, va adquirir a la Universitat de Cambridge el seu sistema de virtualització Hypervisor, realitzat amb programari lliure i anomenat Xen, per 500 milions de dòlars.

Aleshores, Citrix va fer una adaptació del Xen original i el va anomenar XenServer. Aquest consta de versions enfocades a la venda i que tenen moltes funcionalitats, i la versió gratuïta, que és la que avaluarem.

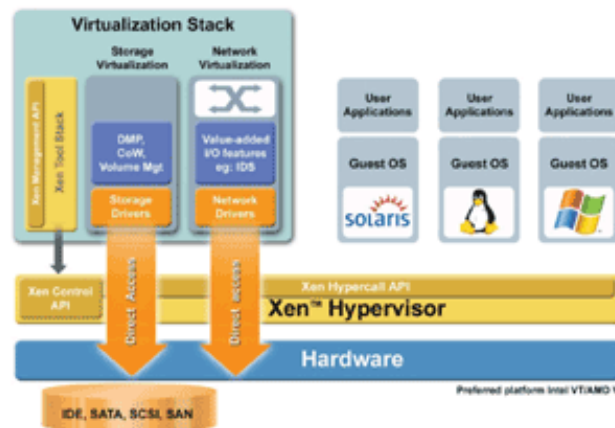
Citrix XenServer permet que les empreses implementin i administrin de manera ràpida i fàcil equips virtuals d'alt rendiment amb Windows i Linux. A més, poden administrar el seu emmagatzematge i els recursos de xarxa des d'una única consola d'administració fàcil d'utilitzar.

La base de la família de XenServer és l'hipervisor de Xen de codi font obert, un motor provat i totalment compatible per a la virtualització de servidors. Amb la virtualització de Xen, una capa prima de programari (coneguda com l'hipervisor de Xen) s'instal·la directament al maquinari, o *bare metall* (amb baix nivell de programació), i s'insereix d'aquesta manera entre el maquinari del servidor i el sistema operatiu. Això ofereix un nivell d'abstracció que permet que cada servidor físic executi un o més "servidors virtuals", separant efectivament el sistema operatiu i les seves aplicacions del servidor físic subjacent. La tecnologia de paravirtualització de Xen és àmpliament reconeguda en el món de la virtualització i millora encara més si s'aprofiten al màxim les últimes capacitats d'assistència a la virtualització del maquinari Intel VT i AMD-V. L'hipervisor de Xen és molt eficient, en part gràcies a la seva simplificació en el seu codi font, d'aproximadament 50.000 línies de codi, fet que es tradueix en un consum fix extremadament baix i un rendiment gairebé nadiu per als convidats.

Citrix XenServer combina el rendiment, la seguretat i l'obertura de la tecnologia de Xen amb l'administració integral de XenCenter, una plataforma perfecta per adoptar ràpidament la virtualització per a consolidació de servidors, continuïtat de l'empresa i desenvolupament i proves de programari. Citrix XenServer és una plataforma nativa de virtualització de 64 bits,

amb la capacitat d'ampliació requerida per les aplicacions crítiques per a l'empresa. Els límits més alts de memòria i CPU *host* i *guest*, sumats al control granular de recursos del CPU, la xarxa i el disc, permeten que ofereixi una qualitat òptima de servei.

La gestió de la plataforma es realitza mitjançant el XenCenter, una aplicació d'escriptori que només funciona en Windows. Cal destacar que hi ha dos projectes per a la creació de plataformes de gestió lliures, com OpenXenManager i XenwebManager (per manejar la plataforma via web). Des d'aquesta aplicació podem afegir servidors que tinguin instal·lat XenServer i gestionar les diferents màquines virtuals que arrencam en cada un d'ells.



Il·lustració N: Esquema funcionament XenServer

Un altre dels detalls interessants de XenCenter és la possibilitat de connectar-se en remot a les consoles o escriptoris de les màquines virtuals, així se centralitza tota l'administració i gestió des d'un únic punt.

Quines característiques ens ofereix aquesta plataforma?:

- Virtualització de sistemes Windows i Linux.
- Sense limitació de memòria assignada a les màquines virtuals.
- Suport per a emmagatzematge compartit en xarxa.
- Migració en calent de màquines virtuals entre nodes (creant-se clústers entre els diferents nodes físics).
- Gestió centralitzada (amb XenCenter) de les màquines virtuals i biblioteca centralitzada d'imatges i plantilles base (entorns i sistemes operatius).

Els requisits que necessita un servidor per poder executar el XenServer són els següents:

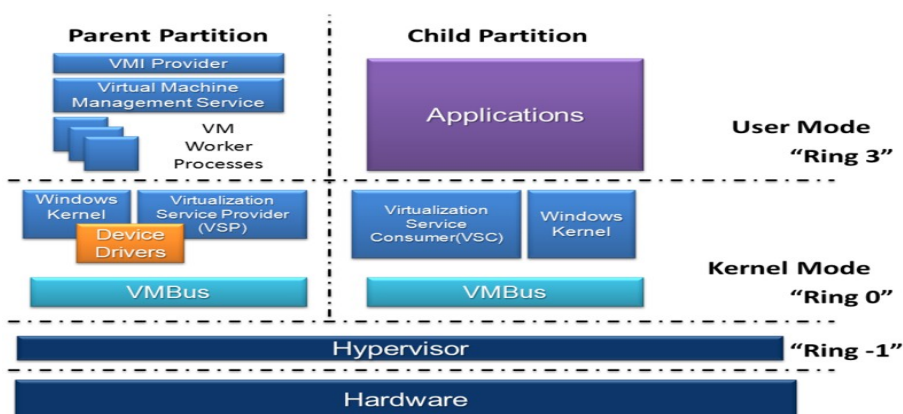
- Processador 64-bit x86.
- CPU: 1.5 GHz mínim, 2 GHz
- Tecnologies Intel® VT or AMD-V™ requerides per a virtualitzar Windows.
- Memòria física de 2GB fins a 1TB.
- Fins a 64 processadors lògics.
- Tarja xarxa de 100Mb/s.
- Fins a 16 NICs.
- Dispositius d'emmagatzematge de diferent tipus com ara: Local, Fibra o SAN.

5.4. Microsoft Hyper-V

La proposta de Microsoft en el món de la virtualització s'anomena Hyper-V, amb el nom en clau de Viridiam, formalment conegut com Windows Server Virtualization. Es tracta d'un sistema basat en un hipervisor per a sistemes de 32 i 64 bits.

La seva solució tant pot ser independent en una màquina sense sistema operatiu anomenat Stand-alone o pot formar part d'un mateix Windows 2008 Server R2.

La versió Stand-alone abans esmentada és gratuïta i és una variant de la instal·lació completa del Windows Server 2008. Aquesta, l'Stand-alone, està limitada a un terminal, sense tenir els afegits que pugui tenir un servidor Windows, ni altres serveis que no siguin els específics de la virtualització. No requereix la instal·lació de cap Windows Server 2008.



Il·lustració 0: Esquema funcionament Hyper-V

Hyper-V admet l'aïllament en termes d'una partició. Una partició és una unitat lògica d'aïllament, amb el suport de l'hipervisor, en el qual executen els sistemes operatius. Un hipervisor, per exemple, ha de tenir almenys una partició primària, que executa Windows Server 2008. La pila de virtualització s'executa a la partició principal i s'hi pot accedir als dispositius físics. La partició dels pares crea les particions secundàries que acullen els sistemes operatius convidats. Una partició pare crea particions secundàries que utilitzen l'API d'Hypercall, que és la interfície de programació d'aplicacions exposades per Hyper-V.

Una partició virtual no té accés al processador físic, ni gestiona les interrupcions reals. Per contra, té una vista virtual del processador i s'executa en adreces virtuals de convidats (*guests*), i depèn de la configuració de l'hipervisor. Un hipervisor podria optar per exposar només un subconjunt dels processadors a cada partició. L'hipervisor s'encarrega de les interrupcions en el processador i la redirigeix a la partició respectiva usant un controlador de lògica d'interrupció sintètic (SynIC). Hyper-V pot accelerar el maquinari de la traducció d'adreces dels clients de direcció d'espais mitjançant l'ús de segona traducció d'adreces del nivell previst per la CPU, conegut com l'EPT d'Intel i XHI (abans TNP) en els processadors AMD.

Les particions fills tenen accés directe als recursos de maquinari i una vista virtual dels recursos, en termes de dispositius virtuals. Qualsevol sol·licitud dels dispositius virtuals es redirigeix a través de la VMBus als dispositius a la partició principal, que gestionarà les sol·licituds. El VMBus és un canal lògic que permet la comunicació entre particions. La resposta també és redirigida a través dels VMBus. Si els dispositius de la partició principal són també dispositius virtuals, seran redirigits fins que arribin a la partició principal, on es tindrà accés als dispositius físics. Les particions de pares consisteixen a executar un proveïdor de serveis de virtualització (VSP), que connecta amb el VMBus i s'ocupa de les sol·licituds d'accés a dispositius de les particions del nen. Dispositius secundaris de particions virtuals: internament executa un client de serveis de virtualització (VSC), el qual redirigirà la petició a la VSP a la partició dels pares a través dels VMBus. Tot aquest procés és transparent per al sistema operatiu convidat.

Els dispositius virtuals també poden prendre avantatge d'una característica de virtualització del Windows Server, amb nom Enlightened I/O, per als subsistemes d'emmagatzematge, de xarxa i gràfics, entre d'altres. Enlightened I/O està especialitzada en virtualització conscient de l'aplicació d'alts protocols de nivell de comunicació com SCSI per aprofitar VMBus directa, que permet passar per qualsevol capa d'emulació de dispositius. Això fa que la comunicació sigui més eficient i que requereix el sistema operatiu convidat per donar suport Enlightened I/O. Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista, Red Hat Enterprise Linux i SUSE Linux es troben actualment entre els únics sistemes operatius que admeten l'Enlightened I/O, cosa que els permet, per tant, córrer més ràpid que els sistemes operatius convidats Hyper-V i que altres sistemes operatius que necessiten utilitzar un maquinari emulat més lent.

5.5. Proxmox VE 2.0

Proxmox Virtual Environment és una aplicació de virtualització basada en programari lliure, concretament un hipervisor de la màquina virtual. Aquesta distribució està mantinguda i desenvolupada per l'empresa Proxmox Solutions GmbH, la qual disposa de suport financer per part de la Internet Foundation Austria (IPA).

Proxmox és una distribució realitzada sota llicència GPL, desenvolupada i mantinguda per l'empresa Proxmox Server Solutions conjuntament amb el suport econòmic d'Internet Foundation Austria. No té cap cost associat, però podem subscriure'ns al manteniment que ens pugui donar la mateixa empresa de pagament.

La instal·lació del Proxmox VE i les seves eines es realitza en un temps aproximat de 5 minuts, depenent, és clar, del maquinari a emprar.

Es tracta d'una solució de virtualització *bare metal*. El terme *bare metal* (metall nu) significa que s'inicia des d'un servidor de buit, que no hi ha cap sistema operatiu anterior.

La distribució del Proxmox VE inclou:

- Sistema operatiu (Debian Squeeze, 64-bit).
- Particionament del disc amb el sistema LVM2. Particions lògiques virtuals.
- Proxmox VE nucli amb suport per OpenVZ (contenidors) i KVM (virtualització total).
- Eines de còpia de seguretat i restauració.
- Interfície de gestió i seguiment a través del web.

- Funcions de clúster com permetre la migració en viu de màquines virtuals d'un servidor físic a un altre. Per fer això necessitam un espai compartit d'emmagatzemament, com ara una solució SAN o DRBD.

Atès que Proxmox VE està basat en Debian, és possible emprar aquesta màquina per altres, per utilitzar-la a partir dels paquets en una màquina existent, sense perdre les seves dades.

5.5.1. Kernel-based Virtual Machine (KVM)

És una infraestructura de virtualització per al Kernel de Linux i OpenSolaris. KVM suporta virtualització nativa allà a on els processadors tinguin les extensions de virtualització disponibles i habilitades.

És una tecnologia que funciona sobre equips que tenen habilitades les extensions BIOS Intel VT o bé les ADM-V.

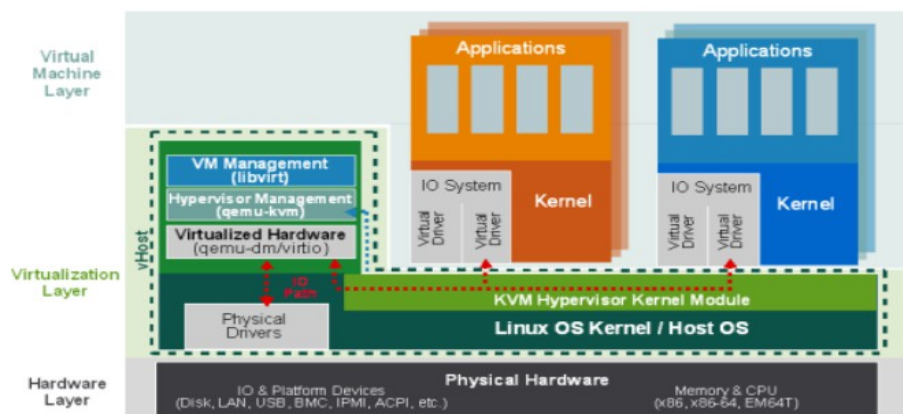


Illustration P: KVM esquema

Es tracta d'una solució completa; per tant, podem crear màquines virtuals de qualsevol tipus que suportin el sistema operatiu que seleccionem.

5.5.2. QEMU

QEMU és un emulador i virtualitzador genèric de codi obert.

Quan s'utilitza com un emulador de màquina, QEMU pot executar sistemes operatius i programes fets per a una màquina (per exemple, una placa ARM) en una màquina diferent. Mitjançant l'ús de la traducció dinàmica (*dynamic translation*), s'aconsegueix un rendiment òptim.

Quan s'utilitza com a virtualitzador, QEMU arriba prop de rendiments nadius executant el codi del convidat (*guest*) directament en la CPU de l'amfitrió (*host*). QEMU suporta virtualització quan s'executa sota l'hipervisor Xen o emprant el mòdul KVM en Linux (com és el cas del Proxmox).

5.5.3. OpenVZ

OpenVZ (virtualització de codi obert) és una tecnologia de virtualització en l'àmbit de sistema operatiu, basat en el Kernel de Linux i sistema operatiu. Permet que un servidor físic executi múltiples instàncies de sistemes operatius aïllats, coneguts com contenidors, servidors privats virtuals (VPS) o entorns virtuals (VEs). És similar a les *jails* de FreeBSD i les "zones" de Solaris.

Aquesta tecnologia comparada amb altres solucions tipus VMware o Xen és una tècnica limitada, ja que requereix que tant el sistema operatiu amfitrió (*host*) com el client (*guest*) hagin de ser Linux. Per contra, el seu rendiment és molt elevat, atès que només perd entre un 1 % i un 3 % de rendiment comparat amb una màquina nativa.

5.6. RHEV 3.0 (Red Hat Enterprise Virtualization)

Red Hat Enterprise Virtualization (RHEV) és una solució de gestió de virtualització completa per a la virtualització de servidors i sistemes d'escriptori, i és la primera plataforma de virtualització empresarial totalment basada en codi obert. Pren com a base l'hipervisor de màquina virtual basat en nucli (KVM) i la plataforma de gestió de virtualització oberta oVirt, ambdós projectes iniciats en Red Hat i oferts a la comunitat de codi obert. RHEV representa una alternativa de virtualització estratègica per a les organitzacions que busquen un menor cost total de la propietat, una rendibilitat més ràpida i una amortització accelerada, així com evitar la dependència d'un únic proveïdor quan es compara amb els proveïdors de virtualització propietaris.

Red Hat Enterprise Virtualization per Servidors inclou el sistema de gestió RHEV Manager i RHEV Hypervisor, a més d'admetre l'ús de sistemes operatius de servidor com els *guests* virtuals.

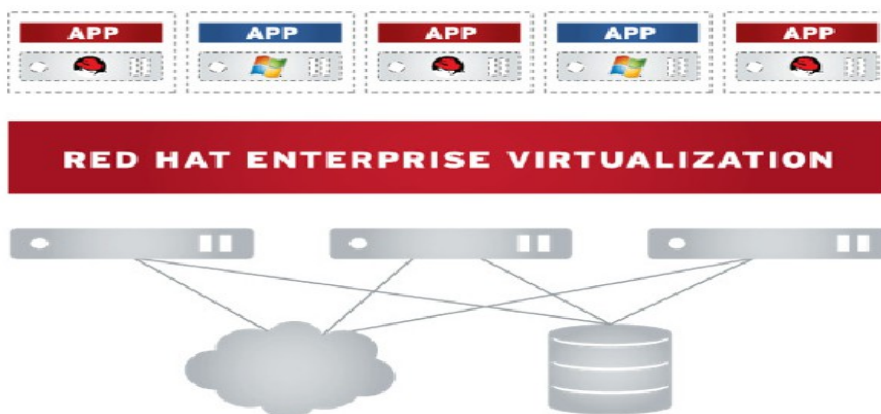


Illustration Q: Esquema virtualització RHEV

Tipus	Descripció
Virtualization Hypervisor	<ul style="list-style-type: none"> • Un hipervisor segur, d'alt rendiment i bàsic basat en el nucli reforçat per a empreses Red Hat Enterprise Linux amb tecnologia KVM (màquina virtual basada en nucli) • Hipervisor de grandària reduïda basat en imatges amb empremta de seguretat minimitzada • Suport per VLAN, unificació de xarxes i una àmplia gamma de dispositius de xarxa, inclosa una interfície de 10 GB • Admet totes les plataformes de servidor x86 de 64 bits que estiguin certificades per Red Hat Enterprise Linux (necessita extensions de virtualització de maquinari Intel-VT o AMD-V). Admet tots els sistemes d'emmagatzematge certificats per Red Hat Enterprise Linux
Administrador de xarxa Hat	<ul style="list-style-type: none"> • Un sistema de gestió centralitzat amb una interfície gràfica que admet fins a centenars de <i>hosts</i> i milers de màquines virtuals (basat en RHEL 6 i JBoss EAP per oferir un rendiment i escalabilitat superiors)
Gestió empresarial	<ul style="list-style-type: none"> • Migració en viu, equilibri de càrrega basat en polítiques, alta disponibilitat, estalvi d'energia, manteniment de clústers, gestió d'imatges, plantilles, aprovisionament lleuger, monitorització d'esdeveniments, entre altres funcions
Rendiment i escalabilitat	<ul style="list-style-type: none"> • Els <i>hosts</i> admeten fins a 160 nuclis i 2 TB de RAM. Els <i>guests</i> admeten fins a 64 vCPU i 512 GB de RAM. Els clústers admeten fins a 200 <i>hosts</i>
Suport tant per a màquines virtuals Windows com Linux	<ul style="list-style-type: none"> • Servidors: Red Hat Enterprise Linux, RHEL 3, 4, 5 i 6; de 32 i 64 bits; Windows Server 2003, 2003 R2, 2008, 2008 R2, de 32 i 64 bits, amb certificació SVVP i WHQL • Escriptoris: Red Hat Enterprise Desktop 5 i 6, 32 i 64 bits; Windows XP de 32 bits; Windows 7 de 32 i 64 bits
Seguretat avançada basada en el Kernel SELinux	<ul style="list-style-type: none"> • Màxim nivell de seguretat basada en el nucli per permetre una detecció immediata de les intrusions i l'aïllament de màquines virtuals i <i>hosts</i> amb la protecció sVirt i Security Enhanced Linux (SELinux)
Automatització i personalització	<ul style="list-style-type: none"> • L'API de RESTful permet automatitzar, gestionar i configurar mitjançant programació tots els aspectes de Red Hat Enterprise Virtualization • Una interfície de línia d'ordres basada en Python permet la programació mitjançant comandes i l'automatització, a més d'haver-se desenvolupat amb la comunitat innovadora • El mecanisme de connexions permet als administradors definir els scripts per modificar definicions de màquines virtuals o executar comandaments del sistema
Portal autoservei d'usuaris	<ul style="list-style-type: none"> • Permet als usuaris finals autoproveir-se de màquines virtuals, definir plantilles i administrar els seus propis entorns
Generació d'informes i monitoritzacions	<ul style="list-style-type: none"> • Les capacitats de generació d'informes històrics detallats, basades en informes Jasper, estan integrades en el producte bàsic per monitoritzar l'ús, les tendències i la qualitat de servei al llarg del temps. S'inclouen més de 25 informes i panells creats prèviament
Requisits del sistema	<ul style="list-style-type: none"> • RHEV-Manager: es recomanen 1-2 processadors x86_64 de nucli quàdruple, 16 GB de RAM, 50GB de disc, NIC Ethernet d'1 GB • Hipervisors RHEV: 1 CPU amb extensions de CPU Intel ® 64 o AMD64 i extensions de virtualització de maquinari AMD-VTM o Intel VT ®. 10 GB de RAM. Emmagatzematge en disc local de 10 GB. Un NIC Ethernet d'1 GB • Consola administrativa: Internet Explorer 7 o superior en Windows amb .Net 4 Framework instal·lat • Portal d'usuaris: Internet Explorer 7 o superior en Windows amb SPICE ActiveX instal·lat o Mozilla Firefox 3.6 o superior en Red Hat Enterprise Linux 5.5 o superior amb SPICE XPI instal·lat

6. Interfícies de treball

Entenem com a interfícies de treball aquelles eines que ens permeten interactuar amb l'usuari i l'ordinador, per realitzar unes tasques determinades.

En aquest cas, allò que avaluarem seran les interfícies de les quals disposa cada plataforma per a la gestió de les màquines virtuals.

Parlarem de la particularitats que puguin tenir, ja que partim de la base que les operacions bàsiques com ara altes, baixes, modificacions, eliminacions de màquines de virtuals, estan més que contemplades, a més d'altres opcions que considerarem bàsiques i no en parlarem, perquè no és l'objecte d'estudi del present projecte.

En parlar d'interfície, en tendrem en compte bàsicament de dues: la que porta el servidor que sustenta les màquines virtuals, que anomenarem *consola*, i la interfície que distribueixen aquestes aplicacions per poder gestionar les màquines virtuals de forma remota, que generalment es tracta d'un client que s'ha d'instal·lar a l'equip client i d'altres que funcionen via web.

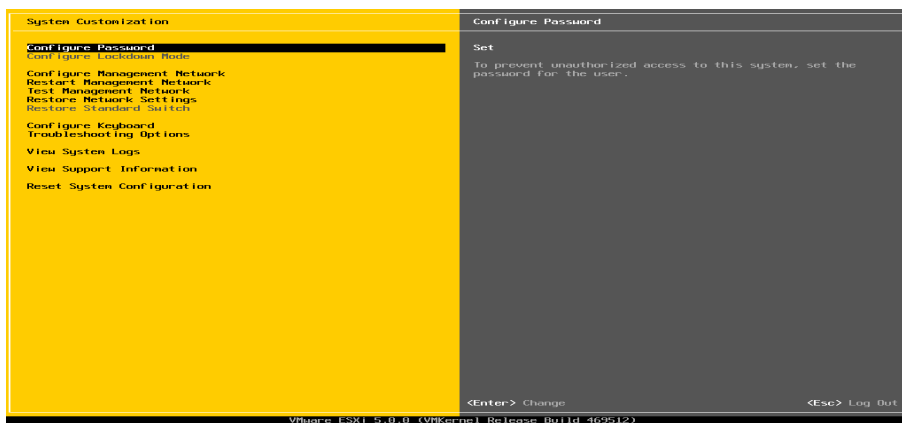
Aquest tema tractarà de les qüestions que hem considerat més interessants de cada solució i no es pretén en cap moment que en sigui cap guia ni manual d'ús.

6.1. VMware ESX 5.0

La interfície del VMware és molt completa i intuïtiva. A mesura que anam llicenciant el producte, s'hi van activant opcions i utilitats que romanen desactivades; per tant, l'eina client només ens l'hem de descarregar un cop.

6.1.1. Servidor

La consola del VMware és una consola de text, amb una sèrie de menús, amb ella no podem fer gaires operacions, de les quals podem destacar:



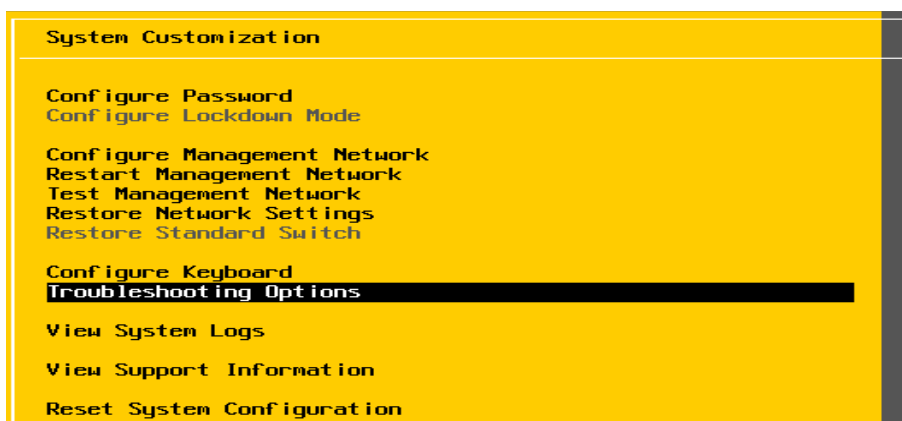
Il·lustració R: vmware ESX - Interfície de la consola

- Configure Manager Network: ens permet configurar les opcions de la xarxa, si volem una IP estàtica o bé dinàmica.
- View System Logs: emmagatzema informació sobre el sistema per a futures consultes, en cas de fallada del sistema o bé per detectar errors o un funcionament ineficient.
- Accés a consola.

Respecte a aquest punt, que sembla molt trivial per a qui està acostumat a emprar programari lliure, a les empreses de programari privat no els sembla una opció molt òptima que els usuaris puguin moure's per l'interior del sistema.

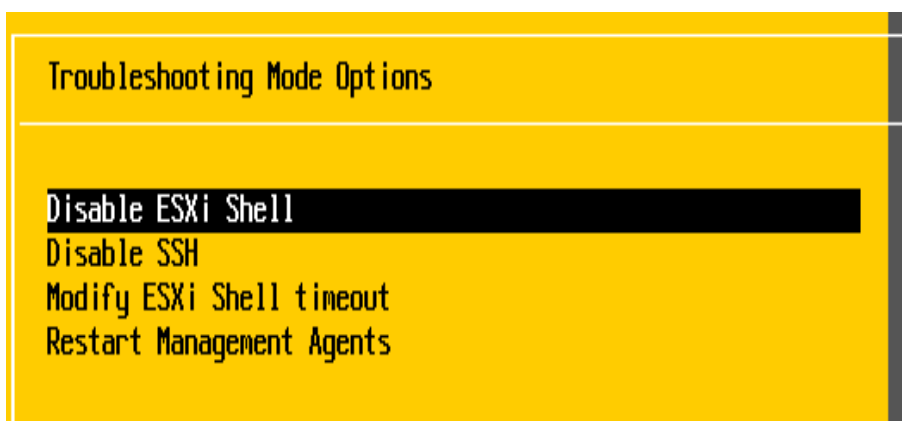
Fins ara VMware tenia una consola oculta i s'havia de fer una espècie de *hack* per poder-hi entrar. En aquesta versió sembla que han canviat un poc la seva ideologia i ara es permet accedir via consola al terminal de comandament i també amb facilitat configurar l'accés via SSH per dur a terme manteniments específics als servidors. En la majoria dels casos no haurem d'emprar-lo, però està molt bé poder tenir control quasi absolut sobre la màquina i el sistema.

Per fer això hem d'anar al menú amb un nom gens intuïtiu anomenat "Troubleshooting options".



Il·lustració S: vmware ESX - detall menú "troubleshooting options"

Un cop a dins, sí que veim clarament les opcions d'“Enable ESXi Shell” i “Enable SSH”.



Il·lustració T: vmware ESX - Detall habilitació shell / SSH

Un cop habilitats, des de la consola empram les combinacions de tecles ALT + F1 i ja podrem accedir directament a consola. El mateix nivell d'accés tendrem si ho feim via SSH (Secure Shell).

```

ESXi 5.0.0 http://www.vmware.com
Copyright (c) 2007-2011 VMware, Inc.

localhost login: root
Password:
The time and date of this login have been sent to the system logs.

VMware offers supported, powerful system administration tools. Please
see www.vmware.com/go/sysadmintools for details.

The ESXi Shell can be disabled by an administrative user. See the
vSphere Security documentation for more information.
~ # _

```

Il·lustració U: vmware ESX - Demostració terminal del servidor

Per tornar al menú textual fem la combinació ALT – F2.

6.1.2. Client

El client de VMware necessita per funcionar una plataforma Windows, encara que fa molts d'anys que usuaris de tot el món sol·liciten un client natiu per a Linux via fòrums oficials de VMware, l'empresa no té cap objectiu de crear-ho. Per tant, ja hem de saber que d'entrada necessitarem una màquina amb una llicència de Microsoft.

La manera més fàcil per instal·lar-ho és adreçar-se via web a l'ESXi que hem instal·lat, i veurem una web a on una opció és la de la descàrrega del client natiu per a Windows. No obstant aquest enllaç, al contrari que les versions anteriors apunta a la web de VMware, entenem que per disposar així sempre de la darrera versió actualitzada del client.



Il·lustració V: vmware ESX - Detall del link descarrega client

Un cop descarregat, s'instal·la de manera tradicional i sense cap complicació.

És una curiositat esmentar que la grandària del client és major que la plataforma de virtualització; el client ocupa 350 MB, mentre que el VMware ESX té una grandària de 305 MB.

La interfície com a eina de gestió és de les millors que podem trobar, té control de tot el

sistema, tant de màquines virtuals com espais d'emmagatzemament de forma molt fàcil de manejar.

Cal destacar també que hi ha apartats de gestió i control de la màquina física a on es troba instal·lat el sistema, així com de moment les millors estadístiques a tot nivell del que està succeint en el sistema, tant en l'àmbit de servidor físic com de màquines virtuals.

Cal destacar també la facilitat de crear alarmes per a esdeveniments concrets, com ara elevada temperatura, ús de CPU, ús de RAM i un llarg etcètera.

Direm també que podem rebre les alarmes i esdeveniment variats a un compte de correu determinat, però com a penalització greu direm que la configuració del client de correu de VMware no permet configurar comptes de correu que requereixen autenticació. Així, necessitarem un servidor de correu que no requereixi autenticació, i avui en dia és una situació bastant poc freqüent.

En aquesta pantalla tenim una visió del vSphere Client 5.0, hi podem veure el que seria la màquina física i organitzada en arbre. Hi podem observar quatre màquines virtuals i a la finestra de la dreta, les característiques en aquest cas de la màquina seleccionada i les pestanyes a on està agrupat per tipus diferents tasques que feim amb elles. Hem de dir que les pestanyes estan col·locades de tal manera que les més emprades van d'esquerra a dreta, la qual cosa facilita el seu ús.

The screenshot displays the vSphere Client interface for a virtual machine named 'Debian'. The main content area is divided into several sections:

- General:** Guest OS: Debian GNU/Linux 6 (64-bit), VM Version: 8, CPU: 1 vCPU, Memory: 1024 MB, Memory Overhead: 67,18 MB, VMware Tools: Not running (Not installed), IP Addresses: (empty), DNS Name: (empty), State: Powered Off, Host: localhost.localdomain, Active Tasks: (empty), vSphere HA Protection: N/A.
- Resources:** Consumed Host CPU, Consumed Host Memory, Active Guest Memory, Provisioned Storage: 17,11 GB, Not-shared Storage: 16,00 GB, Used Storage: 16,00 GB. A storage table shows 'datastore1' with a capacity of 195,00 GB.
- Commands:** Power On, Edit Settings.
- Annotations:** Notes: (empty).

The Recent Tasks table at the bottom shows the following data:

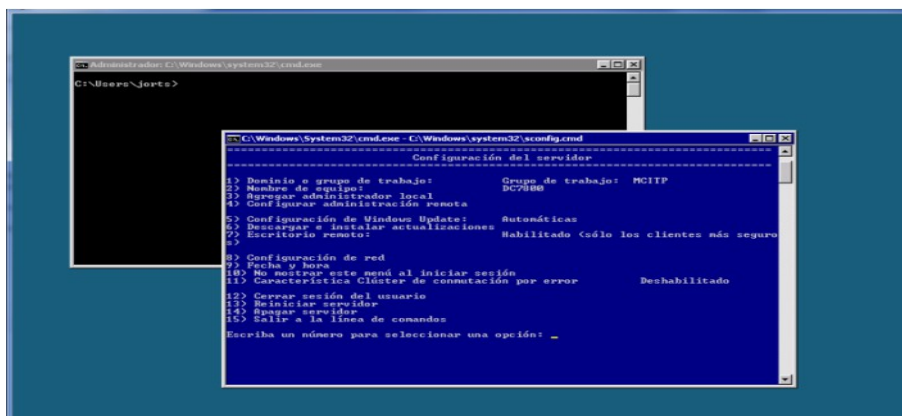
Name	Target	Status	Details	Initiated by	Requested Start Time	Start Time	Completed Time
Create virtual machine	192.168.0.204	Completed		root	19/05/2012 18:09:34	19/05/2012 18:09:34	19/05/2012 18:09:35
Create virtual machine	192.168.0.204	Completed		root	19/05/2012 18:09:21	19/05/2012 18:09:21	19/05/2012 18:09:21
Create virtual machine	192.168.0.204	Completed		root	19/05/2012 18:09:07	19/05/2012 18:09:07	19/05/2012 18:09:07

Il·lustració W: vmware ESX - Vista general de la interfície client

6.2. Microsoft Hyper-V 2008 R2

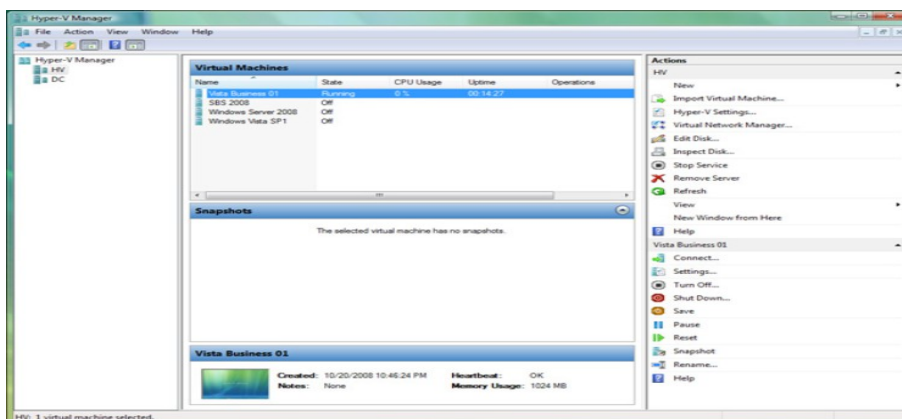
L'hipervisor, que és la solució que ens presenta Microsoft, es pot instal·lar de dues maneres distintes.

Una manera és el tipus Stand-alone, a on només s'instal·la l'hipervisor i el servidor es queda dedicat únicament i exclusivament a aquest servei, així com per exemple les solucions de VMware i Xenserver.



Il·lustració X: Hyper-V Detall versió "stand-alone"

L'altra manera de fer-ho és instal·lar l'hipervisor sobre un Windows Server 2008 R2 de 64 bits, com si d'un servei més es tractàs.



Il·lustració Y: Hyper-V interfície de control sobre Windows 2008 R2

En aquests cas tant consola com client podríem dir que és el mateix, ja que l'hipervisor es controla des de l'Hyper-V manager, a qual també es pot accedir des de l'eina de Microsoft per administrar servidors.

Microsoft disposa d'una eina que ens permet gestionar servidors Windows de forma remota anomenada "Eines d'administració remota del servidor per a Windows 7", aquest programari s'ha d'instal·lar sobre un client amb Windows 7:

<http://www.Microsoft.com/downloads/es-es/details.aspx?familyid=7d2f6ad7-656b-4313-a005-4e344e43997d&displaylang=es>

La interfície de treball està bastant ben aconseguida, de fet aquests tipus d'interfícies són totes bastant semblants i les utilitats oferides no varien gaire les unes de les altres.

Ara bé, un avantatge de la interfície gràfica és que tot està a pocs clics del ratolí, les opcions estan dissenyades de tal manera que no hem de fer excessius moviments per accedir a quasi totes.

Una característica a destacar que experimenta és la velocitat en què es fan els *snapshots*, aquests són quasi al moment, i posteriorment ens dona l'opció de modificar-ne el nom. Les altres solucions ho demanen abans de fer, encara que posteriorment també podem modificar les descripcions.

Una eina que també hem vist que no tenen els altres és un administrador de discos, que ens permet compactar-los a mesura que aquest es van desfragmentant. Encara que és una bona eina, dubtam que realment ho sigui perquè l'HyperV empra el sistema de disc NTFS, mentre que els altres utilitzen sistemes tipus EXT4, VMFS o el sistema propietari de VMware.

Un problema un poc greu que hem vist en aquesta solució és el tema de xarxa, com bé es pot apreciar a les gràfiques. Quan s'instal·la un sistema, es configura aquest mitjançant una targeta de xarxa anomenada *legacy* i posteriorment, un cop instal·lats els *drivers* pertinents, sí que podríem instal·lar una tarja de xarxa adient al sistema operatiu virtualitzat; però, en el nostre cas, a la Debian no va ser possible. Entenem, això no obstant, que un cop instal·lat el *driver* adient, els resultats serien molt semblants als altres productes comparats.

Per contra, diríem que no hem detectat que es puguin configurar alarmes de cap tipus; igual que les estadístiques d'ús, són inexistents.

6.3. Proxmox 2.0

El Proxmox és l'única solució 100 % basada en programari lliure. Sabem que pel nostre compte podem muntar el KVM i el QEMU, i fer anar les màquines virtuals. Però, aquesta gent ens presenta un producte ja consolidat en si mateix amb moltes funcionalitats resoltes i algun afegit molt interessant. Per ser més específics, estam parlant d'una distribució Debian, sobre la qual s'han muntat el KVM i el QEMU a una interfície remota per poder gestionar les màquines virtuals.

6.3.1. Servidor

Un cop iniciat el programari Proxmox, veim a la pantalla del servidor que disposam d'una consola estàndard de GNU-Linux per entrar al sistema. Proxmox empra com a plataforma de la seva virtualització la distribució Debian. Els avantatges d'emprar Debian com a fonament en són uns quants.

1. És una distribució de GNU-Linux consolidada i fa molts d'anys que ja es troba en el món informàtic, des de 1993, per ser més exactes.
2. No és depenent de cap empresa, està desenvolupada per una comunitat de treballadors.
3. Moltes altres distribucions empen Debian com a base per als seus desenvolupaments, com ara Canonical amb Ubuntu.

Sobre aquest sistema 100% basat en programari lliure, l'empresa Proxmox Server Solutions GmbH, s'hi ha decidit per muntar-hi el sistema compost per KVM + Qemu com a solució de virtualització de màquines virtuals, també es tracta de solucions de programari lliure.

```
-----
Welcome to the Proxmox Virtual Environment. Please use your web browser to
configure this server - connect to:

https://192.168.0.205:8006/
-----

Debian GNU/Linux 6.0 auncanada tty1
auncanada login: root
Password:
Last login: Sun Jun  3 10:36:07 CEST 2012 on tty1
Linux auncanada 2.6.32-11-pve #1 SMP Wed Apr 11 07:17:05 CEST 2012 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@aucanada:~# _
```

Il·lustració Z: Proxmox - Interfície terminal

La consola com a tal és del tipus CLI (*command-line interface*), és a dir, una interfície bàsica en mode text a la qual anam introduint les instruccions que volem que es vagin executant via teclat, o per automatitzar-ho, es poden fer scripts o lots d'instruccions.

L'avantatge d'aquest tipus d'interfícies amb la conjunció d'estar implementat Proxmox sobre programari lliure és que podem arribar a manipular o gestionar la màquina principal a un nivell de detall inimaginable. També tenim un major risc de realitzar algun procediment erroni amb nefastes conseqüències.

Per contra, cal dir que la corba d'aprenentatge d'aquestes interfícies és més elevada que en les que consten de menús, ja siguin textuals o gràfics. L'avantatge en aquests cas el tindrà l'administrador de sistemes o professional encarregat de la seva gestió que ja estigui experimentat en el món del GNU/Linux.

Per exemple, podem accedir amb facilitat directament als llocs, *storage* segons la nomenclatura del Proxmox, a on es troben les màquines virtuals.

En aquest cas, hi podem veure 3 màquines 100, 101, 102, que corresponen als identificadors que els varem donar quan els creàrem.

```

root@aucanada:/var/lib/vz/images# ls -l
total 12
drwxr-xr-x 2 root root 4096 May 20 00:10 100
drwxr-xr-x 2 root root 4096 May 20 00:10 101
drwxr-xr-x 2 root root 4096 May 20 00:12 102
root@aucanada:/var/lib/vz/images# _

```

Il·lustració AA: Proxmox - Terminal servidor ubicació VM

Podem veure en aquesta captura de pantalla del client a quina màquina correspon cada ID. Queda palès que manejar des de consola implica tenir una llista ben clara de quina és cada màquina; si no, no sabrem realment quina estem manipulant.

Proxmox Virtual Environment
Version 2.1-1/f9b0f63a

Server View | Datacenter

Datacenter

- aucanada
 - 100 (DebianPRX)
 - 101 (Win2003)
 - 102 (win2008)
 - local (aucanada)

Type	Description	Disk usage	Memory us
node	aucanada	15.0%	19.2%
qemu	100 (DebianPRX)	0.0%	
qemu	101 (Win2003)	0.0%	
qemu	102 (win2008)	0.0%	
storage	local (aucanada)	1.6%	

Il·lustració BB: Proxmox - relació ID amb nom de màquina

6.3.2. Client

El client del Proxmox està realitzat sobre un servidor Apache i amb tecnologia Python i Javascript. A part de l'apunt tecnològic allò que volem donar a entendre és que podem gestionar en el sistema de virtualització emprant com a client un navegador web. Per tant, en aquesta comparativa ens trobam amb el primer i únic client que funciona sobre tres capes, i és independent de la plataforma. Un gran exemple del que hauria de ser el programari en general i no estar fermats a determinades marques comercials.

Type	Description	Disk usage	Memory usage	CPU usage	Uptime
qemu	100 (DebianPRX)	0.0%			-
qemu	101 (Win2003)	0.0%			-
qemu	102 (win2008)	0.0%			-
storage	local (aucanada)	1.6%			-

Start Time	End Time	Node	User name	Description	Status
May 20 00:12:24	May 20 00:12:24	aucanada	root@pam	VM 102 - Create	OK
May 20 00:10:56	May 20 00:10:56	aucanada	root@pam	VM 101 - Create	OK
May 20 00:10:30	May 20 00:10:30	aucanada	root@pam	VM 100 - Create	OK
May 19 22:33:39	May 19 22:33:39	aucanada	root@pam	Start all VMs and Containers	OK

Il·lustració CC: Proxmox - Visió general gestió via web

És la interfície que més s'assembla al Vmware. No és tal volta d'una gran riquesa gràfica, però comparada amb la versió anterior d'aquest producte ha fet un bot qualitatiu molt important.

Podem fer exactament el mateix que a les versions de pagament, fins i tot té el gran avantatge que ens permet fer alta disponibilitat (HA) de servidors, és a dir, poder crear un clúster de servidors que donin servei a les màquines virtuals per repartir la càrrega i, en cas de romputa d'un dels nodes, la resta repartir la tasca del node que ha abandonat.

Aquesta opció, l'HA, als altres productes implica comprar llicències per poder aconseguir tenir el dit recurs habilitat.

Un altre avantatge que té el Proxmox és que via SSH podem accedir a totes les opcions del servidor, i com que es tracta de programari lliure, no trobarem cap impediment per fer i desfer el que vulguin, sota la nostra responsabilitat, és clar.

Un inconvenient detectat és que, a l'hora de realitzar *snapshots* o *backups*, com també els anomenen, resulta que la màquina virtual ha d'estar aturada. No es poden fer en calent.

6.4. Red Hat Enterprise Linux-RHEL

La versió de virtualització de què disposa l'empresa de Red Hat es diu RHEV, Red Hat Enterprise Virtualization.

L'entrega del producte ens recorda a la de Microsoft, hi ha dues opcions: poder disposar del RHEV de forma autònoma (Stand-alone) o muntar un servidor RHEL i sobre ell activar el servei de virtualització.

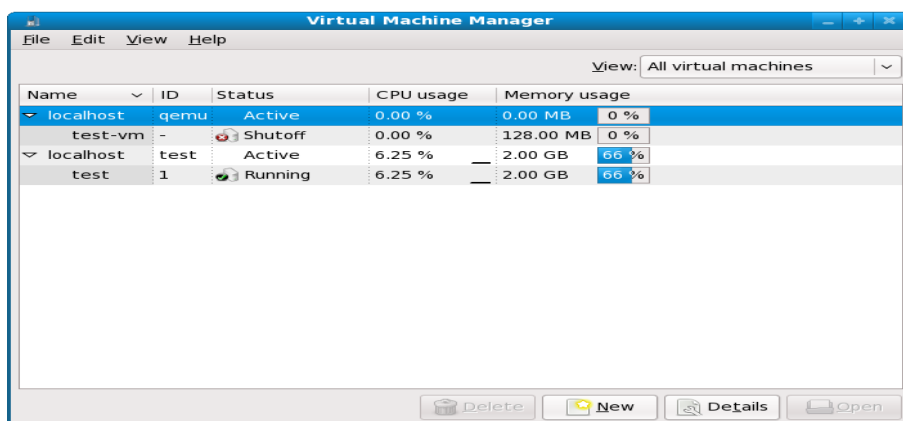
Al nostre cas seleccionarem la segona opció, ja que la versió Stand-alone, varem ser incapaços de fer-la funcionar.

El RHEV per la part tecnològica ens recorda molt al Proxmox, ja que ambdós es basen sobre programari lliure i la plataforma KVM per realitzar la virtualització de les màquines. No obstant això i de forma sorprenent, com hem vist a les gràfiques, el rendiment en ocasions és bastant diferent l'un de l'altre.

6.4.1. Servidor

La consola, com hem dit abans, és semblant només quant a ideologia a Microsoft, és a dir, tenim un servidor com a tal de Red Hat instal·lat i sobre ell feim executar el sistema de virtualització, KVM i QEMU en aquest cas.

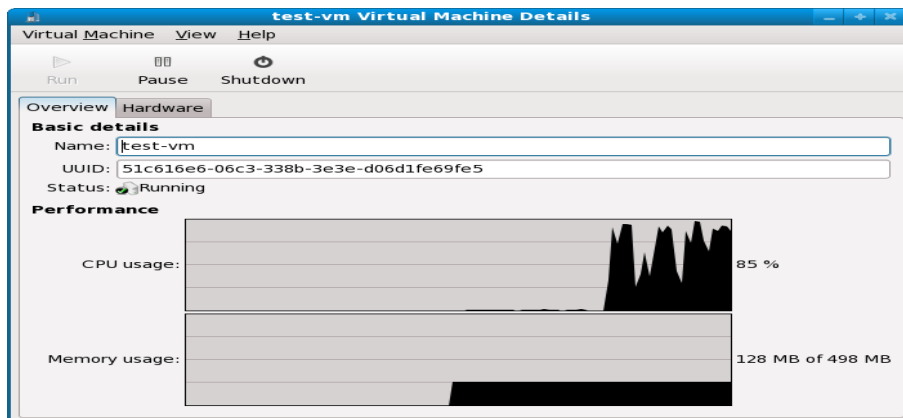
Un cop a dins, executam el mànager del Red Hat, és l'anomenat *virt-manager* i és una eina genèrica per manejar màquines virtuals.



Il·lustració DD: RHEL - virt-manager, una gestió per defecte

Podríem dir que per ara és la interfície és més senzilla que hem vist. Fa la seva funcionalitat, però sense gaires pretensions. No hem trobat cap opció per poder fer *snapshots* de les màquines ni per poder establir-hi alarmes.

Això sí, disposa d'estadístiques de rendiment d'ús sobre les màquines, i hi podem veure l'ús de la CPU i de la RAM.

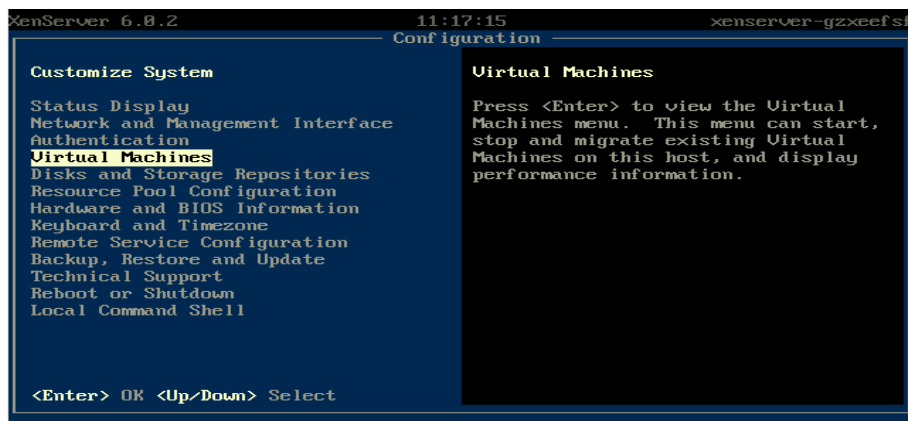


Il·lustració EE: RHEL - estadístiques de màquines virtuals.

6.5. Citrix Xen Server 6.0

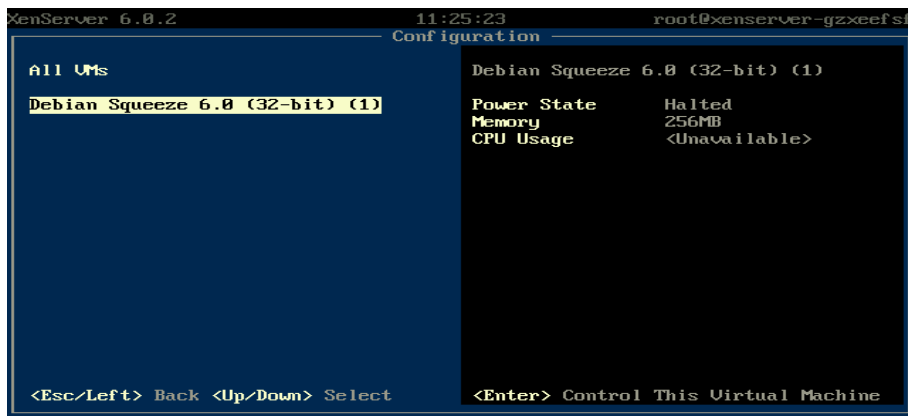
6.5.1. Servidor

Podrem dir que la consola del Xen Server des del nostre punt de vista és la millor que hem vist. Consisteix, igual que el Vmware, en un menú textual que ens permet realitzar moltes operacions directament i sense haver d'anar al terminal a picar instruccions.



Il·lustració FF: XenServer - Menú gestió VM des de consola

A més de tenir les mateixes opcions que el VMware, en té una de molt interessant, que és la de "virtual machines". Des d'ella podrem manipular algunes opcions de la màquina, bàsicament aturar i posar en marxa.



Il·lustració GG: XenServer - Detall gestió VM Debian des de consola

6.5.2. Client

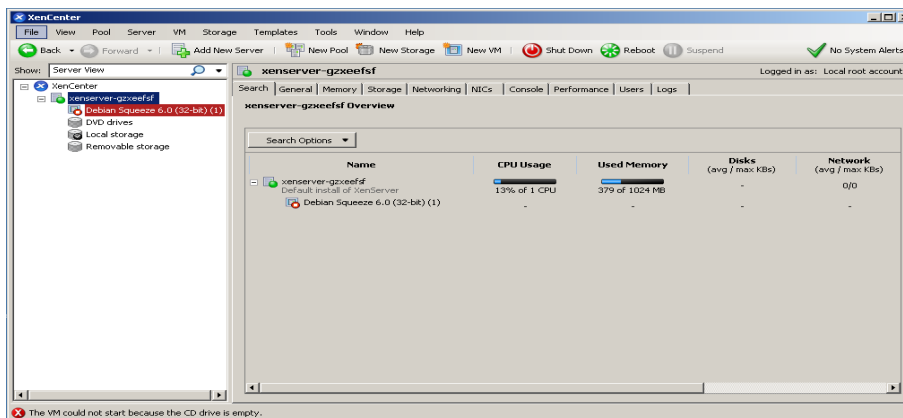
El client segueix la línia de quasi totes les altres solucions, és a dir, es tracta d'un client natiu per al sistema operatiu Windows. El funcionament en aquest cas també és molt semblant al del Vmware. Així, un cop arrencat el servidor, ens hi dirigim amb un navegador a la seva IP i veim un menú, molt breu, a on tenim l'opció de descarregar el client.



Il·lustració HH: XenServer - descarrega del client via web

En aquest cas es fa en mode local; per tant, és molt ràpida la seva descàrrega, a part que la seva grandària només és de 47MB. La duració de la seva instal·lació tarda menys de 5 minuts.

En aquest cas, des del client, hi podem trobar totes les operacions necessàries per a la gestió de les màquines virtuals i dels espais d'emmagatzemament.

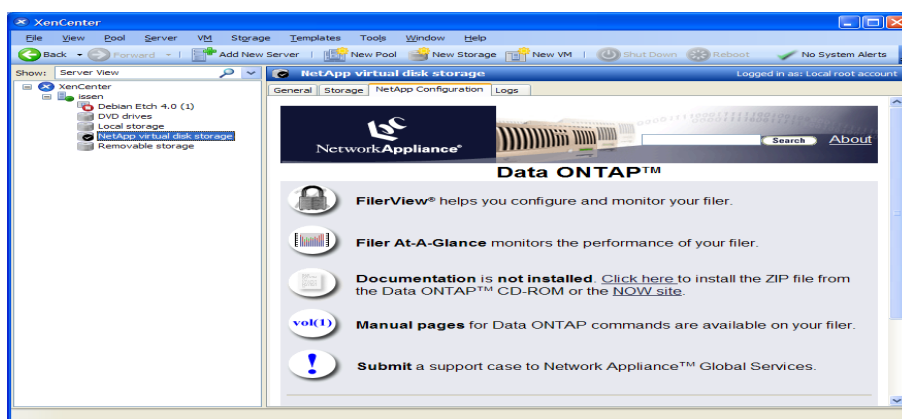


Il·lustració II: XenServer - Visió general eina gestió VM

Podem veure que el client també és molt semblant al del VMware, tant en gestió de màquines com en complexitat d'estadístiques i alarmes del sistema, tant sobre la màquina *host* com de les màquines virtuals i sistema en general.

Un tema interessant que té XenServer i no hem vist a les altres aplicacions és el fet que disposa d'una opció per instal·lar *plug-ins* (connectors). Així s'aconsegueix adaptar el nostre client i expandir les funcionalitats amb molta facilitat i no haver d'estar esperant una actualització generalitzada de tot el client.

Per exemple, podem veure que, gràcies a un de tants *plug-ins* que hi trobam, n'hi ha un que habilita una interfície GUI via web, encara que un cop més el navegador allà on funcioni ha de ser una plataforma Windows; així, no li acabam de veure gaire bé el sentit.



Il·lustració JJ: XenServer - Plug-in per a la gestió via web

L'altra part bona dels *plug-ins* és que permet participar en la seva comunitat. Per tant, a mesura que aquesta plataforma es vagi implantant sorgiran més desenvolupadors que participaran en la creació de *plug-ins*. És un punt molt interessant i nou que pot donar molta vida a aquest projecte.

Una de les dificultats és que per poder disposar d'alarmes al sistema haurem d'adquirir una llicència per a una versió superior. Així doncs, no disposam d'alarmes a la versió gratuïta del producte.

7 Benchmarks

7.1. Introducció

Un *benchmark* és una tècnica que s'empra per mesurar el rendiment d'un sistema o component seu. D'una manera més casual podríem dir que és el resultat de l'execució d'un programa informàtic o conjunt de programes en una màquina, amb l'objectiu d'avaluar el rendiment d'un element en concret i poder comparar-lo *a posteriori* amb màquines similars.

Per al desenvolupament dels *benchmarks* o tests de les plataformes a avaluar, es va arribar a la conclusió que la millor manera de fer-ho era veure quin seria el rendiment d'un servidor virtualitzat idèntic, sobre les diferents plataformes de virtualització, comparat amb el mateix sistema sobre el mateix maquinari sense virtualitzar.

Un cop repetits tots els *benchmarks* a cada una de les màquines virtualitzades sobre les diferents plataformes, tindrèm aleshores una valuosa quantitat d'informació per poder contrastar i extreure opinions sobre els sistemes avaluats.

7.2. Maquinari/programari sistema operatiu bàsic

El maquinari emprat realment no és vinculant als resultats, ja que cada solució es testearà sobre el mateix maquinari. No obstant això, comentarem que es tracta d'un ordinador amb CPU Intel Quad Core, amb 4GB de RAM i una tarja de xarxa de 1000 MB.

Les màquines virtualitzades tindràn la següent configuració estàndard: CPU amb dos *cores*, 2GB de RAM, un disc dur de 5GB i una tarja de xarxa de 100 MB.

Per realitzar les proves amb el sistema natiu deshabilitarem mitjançant instruccions del sistema operatiu dos dels *cores* i desconnectarem físicament 2GB de RAM, a efectes que les màquines siguin al més semblant possible entre elles.

El sistema operatiu sobre el qual es faran les proves serà una distribució de GNU-Linux, en concret la Debian 6.0 de 64bits.

Per tenir un referent del rendiment del sistema, allò que faríem seria instal·lar primer aquest sistema operatiu sobre el maquinari i realitzar tots els tests necessaris. D'aquesta manera, els resultats obtinguts constaran com a referència per a les proves posteriors de les màquines virtualitzades amb les mateixes característiques que aquestes.

Un cop que tenim instal·lades totes les plataformes virtuals amb les seves respectives màquines virtuals a dins elles, procedirem a realitzar els tests de rendiment d'aquestes, amb l'objectiu de poder comparar-les i extreure les conclusions de les dades quan disposem d'elles.

Les proves que es realitzaran es faran sobre 5 components bàsics dels sistemes informàtics: CPU, RAM, disc, xarxa i sistema.

Test CPU: el test de a CPU ens donarà el rendiment que té la màquina quant a la seva capacitat de càlcul i processament d'instruccions.

Test RAM: en aquest tipus de test allò que s'intenta esbrinar és la velocitat a la qual les instruccions o informació es van movent per la memòria RAM de l'ordinador.

Test disc: avaluarem en aquesta sèrie de tests el rendiment del dispositiu de memòria secundària allà on les dades queden emmagatzemades de forma persistent, es faran tant lectures com escriptures.

Test xarxa: avaluarem la velocitat de transmissió de dades entre dos ordinadors d'una xarxa LAN, tant la velocitat de pujada de dades com la de descàrrega.

Test de sistema: són un conjunt de tests o *benchmarks* que avaluen un sistema de forma conjunta. Són tests específics per provar una aplicació o servei determinat sobre un servidor i veure quin és el seu rendiment, com per exemple poder provar el rendiment d'un servidor web.

Per a la realització dels tests, tenim una sèrie d'opcions molt àmplies. Podríem haver realitzat els tests nosaltres mateixos, però vàrem decidir investigar, i trobarem una *suite* específica per a la realització de *benchmarks* de quasi tot tipus sobre ordinadors. La *suite* està catalogada com a programari lliure i llicenciada sota la versió 3 de la GPL.

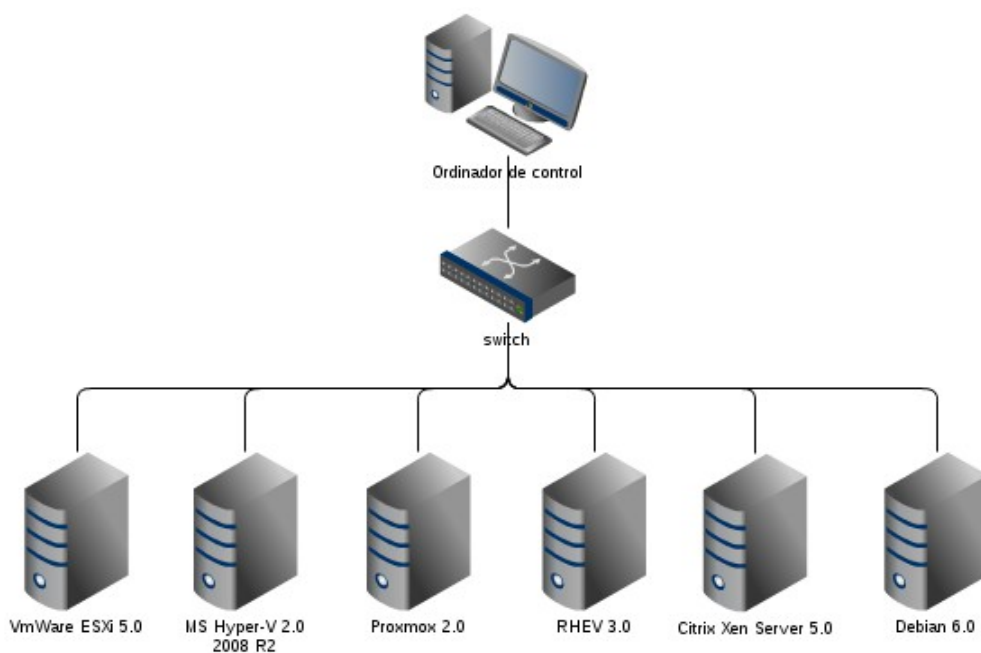
7.3. Disseny del sistema

El sistema que es desenvoluparà consistirà en el següent:

Es crearà un instal·lació per a cada solució de virtualització, VMware ESX 5, Hyper-V, Proxmox 2.0, Red Hat EV 3.0, Xen Server 5.0, a més d'una instal·lació d'una Debian 6.0 sense virtualitzar, que servirà de producte de referència per validar les altres plataformes.

Un cop que tinguem les plataformes instal·lades, a continuació s'instal·larà una màquina virtual amb les mateixes característiques que la Debian sense virtualitzar. Al nostre cas constava de les següents especificacions:

- CPU de 2 cores 1.4Ghz
- 2 MB RAM
- Disc dur de 5GB
- Targeta de xarxa de 100 MB



Il·lustració KK: Laboratori amb plataformes de virtualització

7.4. Eines de *benchmarks* emprades

7.4.1. Phoromix Test Suite

Per realitzar la sèrie de tests sobre els servidors virtuals, emprarem una eina de *benchmark* molt estesa.

La *suite* anomenada Phoromix test *suite*, d'ara endavant PTS, és una *suite* lliure, realitzada sobre codi obert i distribuïda sota llicència GPL v3. Està especialitzada en el testeig del rendiment de maquinari i programari.

És una eina amb bastant de renom mundial i està essent emprada per moltes empreses com a referent dels mesuraments dels seus productes.

Algunes característiques que té el PTS:

- Disposa aproximadament d'uns 130 tests individuals i uns 60 de *suites* o grups.
- Els resultats obtinguts arran dels tests és poden pujar automàticament a una web a on s'emmagatzemen tots els tests realitzats, de tal forma que podem comparar a nivell mundial els resultats obtinguts.
- Es poden fer tests a mida mitjançant programació i després pujar-los a la web de l'entorn col·laboratiu, perquè altres usuaris els puguin emprar.
- Els tests es poden realitzar de forma desatesa (*batch*) i es generen informes dels resultats obtinguts.
- Els resultats obtinguts es poden exportar a diferents formats per a millor manipulació, com ara el format d'intercanvi de dades CVS, el format estàndard de visualització de documents PDF i un llarg etcètera.

La versió que s'ha emprat per realitzar les proves ha estat la 3.8, que en aquests moments era la versió estable.

Els tests de PTS se separen per segments cap allà on està destinat el testeig en concret; així doncs, tenim tests específics per a memòria, disc, CPU i sistema.

7.4.2. Instal·lació *suite*

La instal·lació d'aquesta *suite* en un sistema GNU/Linux no pot ser més senzilla.

Des de consola i amb els permisos de *root*, executam:

```
aptitude install phoronix-test-suite
```

Cal tenir en compte que hem de tenir accés les branques *contrib* de la distribució emprada.

Un cop instal·lat el paquet principal, l'executarem a efectes que s'instal·li correctament, ja que la primera vegada ens farà una sèrie de preguntes que haurem de respondre, com ara si volem participar automàticament en la comunitat a l'hora d'enviar directament els resultats obtinguts per les nostres màquines o tests realitzats.

7.4.3. Scripts PTS

Per instal·lar els tests, que al nostre cas són set, realitzarem un script, ja que la tasca es repeteix.

De la suite PTS hem seleccionat tests de cada modalitat, són els següents:

Típus	Tests emprats
CPU	Compress-7z, gZip
RAM	Stream
Disc	Fmark, Postmark
Sistema	Phpbench, Apache
Xarxa	Test desenvolupat propi

7.4.3.1. Instal·lació: 0install.sh

L'script d'instal·lació era bastant senzill, s'anomenava 0install.sh; el contingut n'era el següent:

```
#instalar samba
aptitude install smbclient smbfs

#processador
phoronix-test-suite install compress-7zip
phoronix-test-suite install compress-gzip

#memory
phoronix-test-suite install stream

#disk
phoronix-test-suite install fs-mark
phoronix-test-suite install postmark

#sistema
phoronix-test-suite install phpbench
phoronix-test-suite install apache
```

Com podem veure instal·lam set test de la *suite* PTS a més de les llibreries Samba, que són les que emprarem per realitzar el test de xarxa i poder-nos connectar a un directori determinat d'una màquina que estigui a la nostra xarxa LAN.

El PTS, a part de descarregar els tests que li hem sol·licitat, també s'encarrega de descarregar totes aquelles dependències o eines auxiliars que necessita per realitzar correctament els tests.

Una vegada instal·lats tots els paquets necessaris, allò que farem serà executar totes les proves mitjançant un altre script i poder deixar la màquina treballar de forma destesa (*batch*).

7.4.3.2. Execució dels tests

En aquest punt el que hem fet és crear un altre script, que es divideix en dues parts; la primera s'encarrega d'executar tots els *benchmarks* (índexs de referència) desitjats; la segona té la tasca de recollir tots els resultats obtinguts i exportar-los a format CVS per a millor tractament amb un full de càlcul com ara l'AbiWord o el calc del LibreOffice.

```
ltest.sh

#processador
phoronix-test-suite batch-run compress-7zip
phoronix-test-suite batch-run compress-gzip

#memory
phoronix-test-suite batch-run stream

#disk
phoronix-test-suite batch-run fs-mark
phoronix-test-suite batch-run postmark
#sistema
phoronix-test-suite batch-run phpbench
phoronix-test-suite batch-run apache

phoronix-test-suite list-saved-results > 1.txt

cat 1.txt | grep Name | cut -c 13-27 > 2.txt

for linea in `cat 2.txt` ; do
    phoronix-test-suite result-file-to-csv ${linea}
done
```

Com podem veure les instruccions que es diuen “batch-run” són les que executen de forma desatesa els *benchmarks* de la *suite*.

La part següent genera un fitxer amb els noms dels resultats obtinguts i posteriorment exportats a CVS per a la seva explotació.

7.5. CPU - Processador

7.5.1. Compress-7-Zip

El test de processador versa sobre l'ús exhaustiu de l'algoritme de compressió 7-Zip. D'aquest breument podem dir que es tracta d'un algorisme de compressió sense pèrdua i amb taxes molt altes, millors que les del ZIP i RAR.

Es tracta d'un algoritme lliure sota llicència Gnu-Igpl.

<http://es.wikipedia.org/wiki/7z>

Aquest test es va realitzar 5 vegades seguides i es va agafar la mitjana dels resultats obtinguts, que es dona en segons de càlcul.

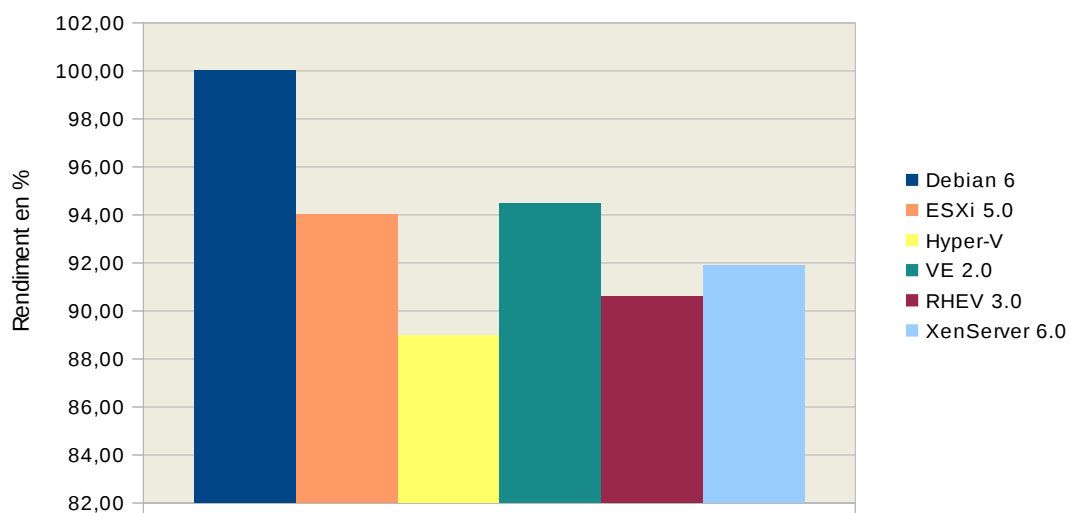
La mitjana de duració per màquina fou d'uns 10 minuts; les sis màquines computaren un total aproximat de 60 minuts a realitzar les operacions.

7.5.1.1. Gràfiques-resultats

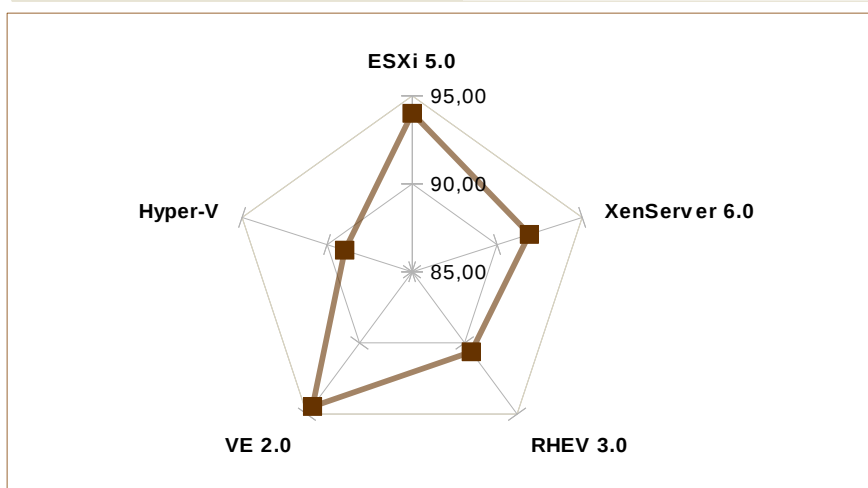
Categoria **Processador** Test **Compress-7zip**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Segons	2837	2667	2524	2680	2571	2607
%	100	94,01	88,97	94,47	90,62	91,89

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració LL: Benchmark processador – compress-7zip

7.5.2. Compressió gZip

El test de compressió gZip és un test de CPU encarregat d'extreure el rendiment del processador mitjançant l'ús exhaustiu de l'algoritme de compressió gZip. És el temps que tarda a comprimir un fitxer de 2 GB de grandària.

El gZip va ser una eina de compressió de dades dissenyada per poder reemplaçar l'altra eina de compressió anomenada Compress. El gZip en comparació tenia molt millor compressió; a més, pot manipular fitxers creats amb l'algoritme Compress, gZip i Pack.

<http://en.wikipedia.org/wiki/Gzip>

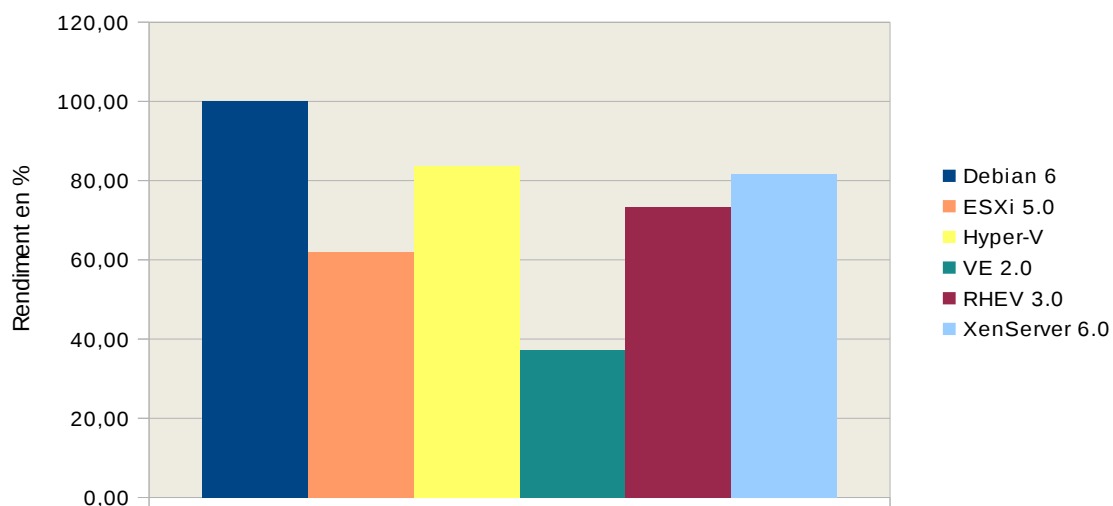
Cal recalcar també que es tracta d'un algoritme distribuït sota codi lliure i llicència GPL.

7.5.21. Gràfiques

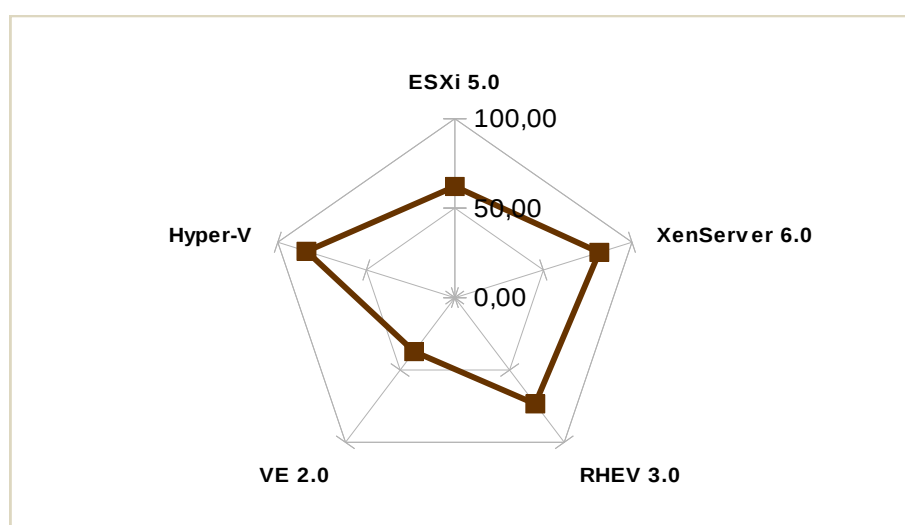
Categoria **Processador** Test **gZip**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Segons	25,77	35,53	29,96	41,93	32,62	30,52
%	100	62,13	83,74	37,29	73,42	81,57

Representació gràfica **Barres**



Representació gràfica **Xarxa**



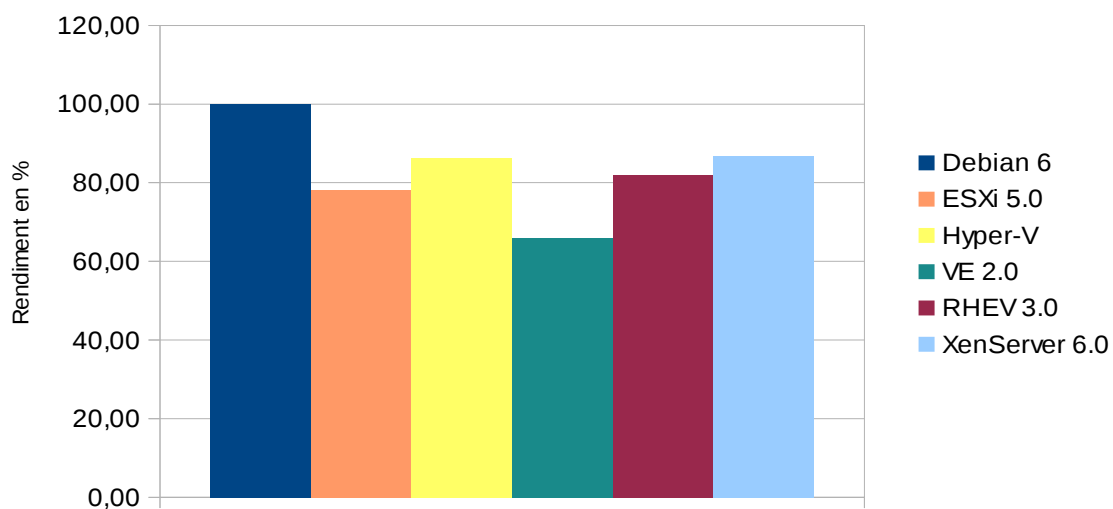
Il·lustració MM: Benchmark processador - gZip

7.5.3. Mitjana dels càlculs

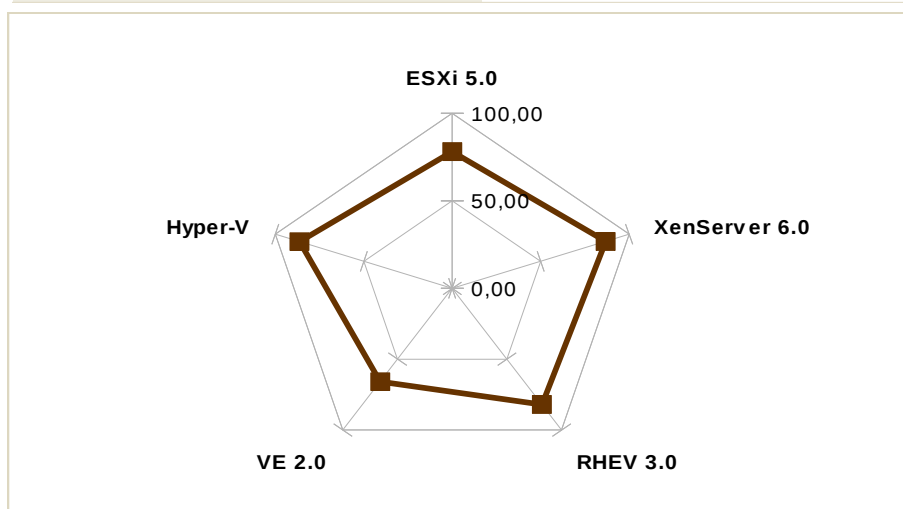
Categoria **Processador** Test **Mitja**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
%	100	78,07	86,35	65,88	82,02	86,73

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració NN: Mitja càlculs resultats processador

7.5.4. Resultat obtinguts

Compress-7-Zip

La mètrica d'aquest test indica que sempre serà millor un resultat de major vàlua que un de menor vàlua.

En aquest cas, podem observar que les màquines virtualitzades en general tenen un rendiment bastant aproximat al rendiment original de la màquina real. Tenim una franja de pèrdua de rendiment d'entre un 6 % i un 11 %, malgrat que de forma visual les diferències siguin molt exagerades, a causa de l'escala emprada en la seva representació.

Els millors resultats en aquest camp, encara que recordem que en global el rendiment és realment òptim, són les solucions de Proxmox i Vmware. Per contra, els resultats obtinguts amb menor rendiment són els de l'Hyper-V de Microsoft, amb un 89 % de rendiment comparat amb la màquina original.

gZip

L'altre test realitzat sobre la CPU dels servidors ens dóna uns rendiments diferents al del 7-Zip. Entenem, doncs, que l'algoritme de compressió del gZip deu ser substancialment diferent al 7-Zip quant a operacions matemàtiques emprades per a la seva finalitat.

Podem veure amb facilitat, gràcies a la gràfica en forma de xarxa, que les solucions que semblen optimitzar millor el tema de CPU en aquest cas són les de VMware i l'Hyper-V de Microsoft. Per altra banda, el Proxmox és el que gestiona pitjor aquest punt.

En aquest test, un menor valor ens indica un millor resultat. Veim aleshores una dada curiosa respecte al test abans realitzat, i és que els que abans eren bons en el test de compressió 7-Zip veim com ara són els que obtenen els pitjors resultats.

Curiosament observem que ara el millor és l'Hyper-V de Microsoft i el pitjor, el Proxmox, quan abans havia demostrat ser un dels millors.

En ambdós casos sembla tenir un resultat més equilibrat la solució aportada pel Citrix Xen Server 6.0.

Entenem, per tant, que les solucions que hi ha al mercat han decidit maximitzar un cert tipus de processos en detriment d'altres als seus hipervisores.

El els casos de Proxmox i Red Hat és sorprenent, ja que l'arquitectura en ambdós és la mateixa, l'única diferència actual és que Proxmox empra una versió de nucli Linux més actualitzada que Red Hat. També cal esmentar que el Proxmox funciona sobre una distribució Debian i el RHEL, sobre una distribució de Red Hat.

7.6. RAM-memòria

7.6.1. Stream

Aquest test comprova el rendiment de la memòria, en concret la seva amplada de banda mesurada en MB/s i la corresponent ràtio de computació per a vectors simples de Kernel.

<http://www.cs.virginia.edu/stream/ref.html>

Software dependències:

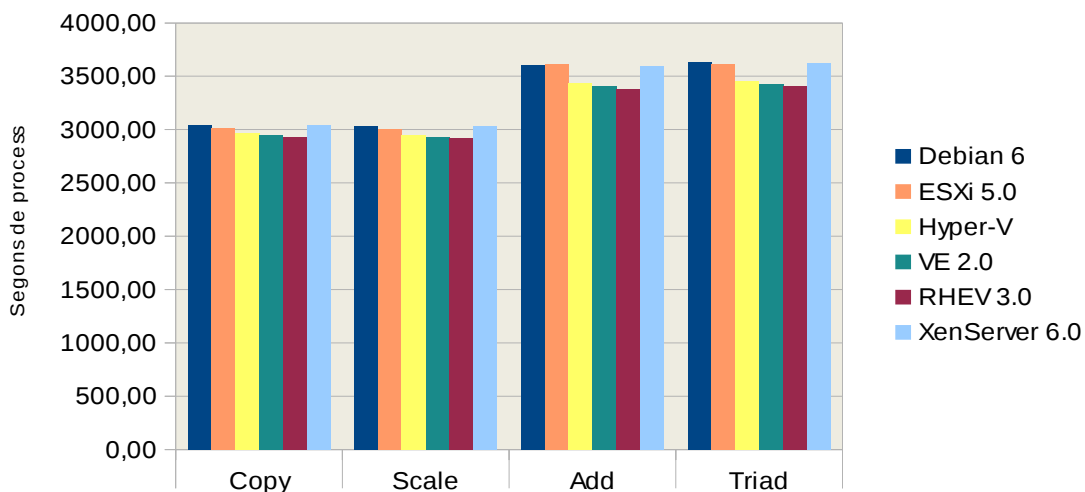
- Compiler / Development Libraries

7.6.2. Gràfiques-resultats

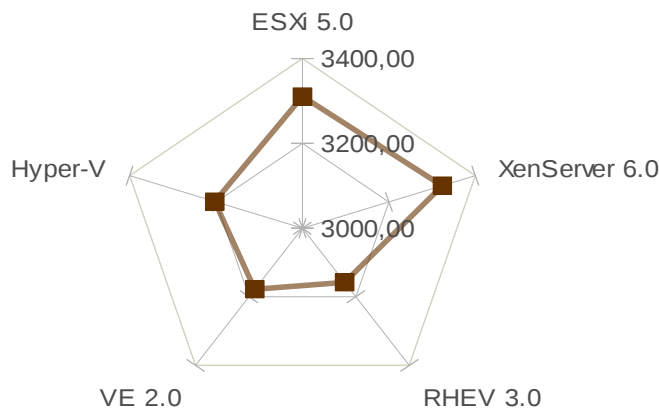
Categoria **Memòria** Test **Stream**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Copy	3046,03	3016,58	2970,96	2944,64	2929,15	3046,67
Scale	3032,37	3002,99	2951,50	2928,68	2917,31	3029,89
Add	3604,37	3610,83	3432,52	3407,80	3381,60	3597,01
Triad	3636,55	3610,48	3455,22	3431,21	3404,23	3620,91
Mitja %	3329,83 100,00	3310,22 99,41	3202,55 96,18	3178,08 95,44	3158,07 94,84	3323,62 99,81

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració OO: Benchmark memòria - Stream

7.6.3. Resultat obtinguts

En aquest apartat podem veure que el rendiment de la RAM en les solucions tant de VMware ESX 5.0 com el Citrix XenServer 6.0 són realment excepcionals, especialment del segon, que té uns resultats quasi iguals que la màquina real.

Per contra, podem veure com les dues solucions que han apostat pel KVM, conjuntament amb l'hipervisor, són les que pitjor resultat donen quant al rendiment de la RAM.

No obstant això, en els resultats purs de la taula en principi no sembla que hi hagi d'haver tanta diferència pel fet de convertir-los en percentatges; veurem les grans diferències.

En línies generals podem concloure que quant a accés a memòria RAM les solucions ofertes donen un rendiment molt proper al que seria la mateixa màquina sense virtualitzar.

7.7. Discs I/O

7.7.1. FS-Mark

Descripció: és un test desenvolupat per provar el rendiment del sistema de fitxers.

En el nostre cas consistia en dues proves, es varen fer els tests sobre 1.000 fitxers d'1 MB cada un; la primera prova es féu de manera normal, i la segona, sobre el mateix nombre de fitxers, però deshabilitant el sistema de sincronització (No sync/Fsync).

Software dependències:

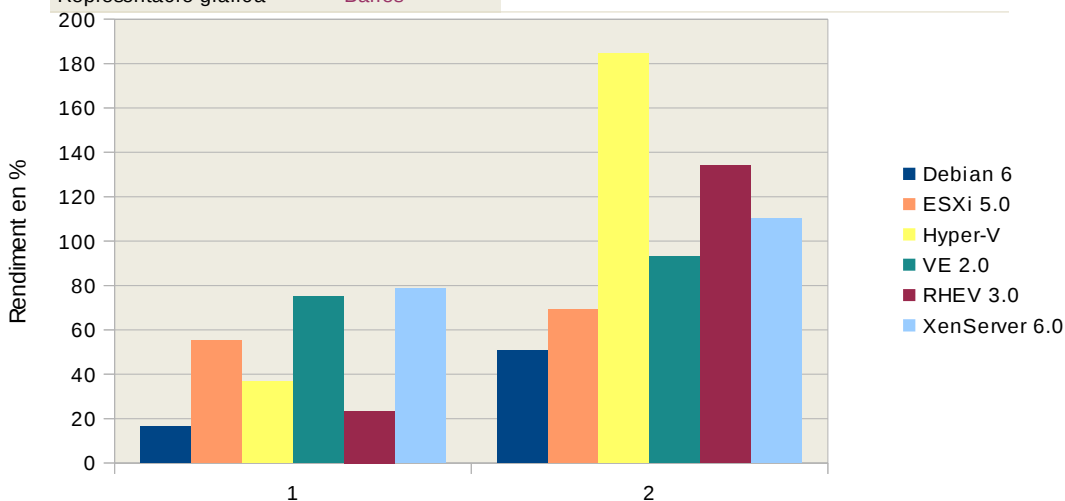
- Compiler / Development Libraries

7.7.2. Gràfiques de rendiment

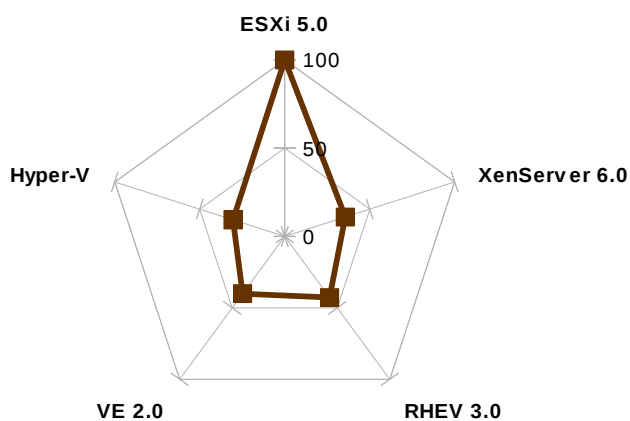
Categoria **Disc** Test **FS-Mark**

Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0	
16,53	55,47	36,83	75,02	23,43	78,57	1000 Files, 1MB Size
50,76	69,07	184,80	93,13	133,87	110,38	1000 Files, 1MB Size, No Sync/FSync
33,645	62,27	110,815	84,075	78,65	94,475	Mitja %
100	100	30,36	40,02	42,78	35,61	

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració PP: Benchmark Disc Test FS-Mark

7.7.3. Postmark

Descripció: consisteix en la prova dissenyada a simular la tasca que reben els servidors web i servidors de correus a base de provar petits fitxers.

El perfil del test realitza 25.000 transaccions amb 500 fitxers simultanis amb grandàries que estan dins el rang de 5 i 512 kilobytes.

Software dependències:

- Compiler / Development Libraries

7.7.4. Gràfiques

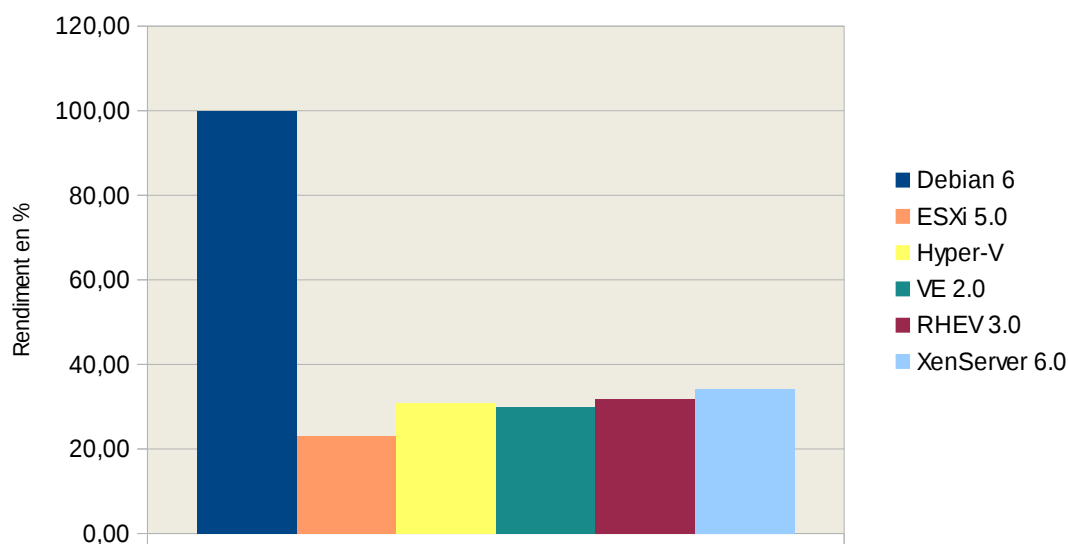
Categoria **Disc**

Test **Postmark**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Mitja	1737,00	399,00	534,00	520,00	551,00	592,00
%	100,00	22,97	30,74	29,94	31,72	34,08

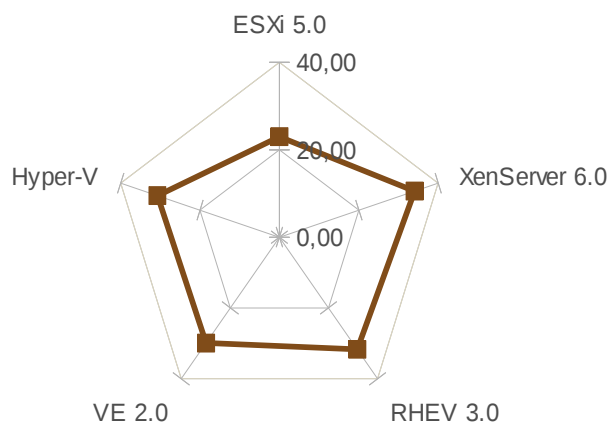
Representació gràfica

Barres



Representació gràfica

Xarxa



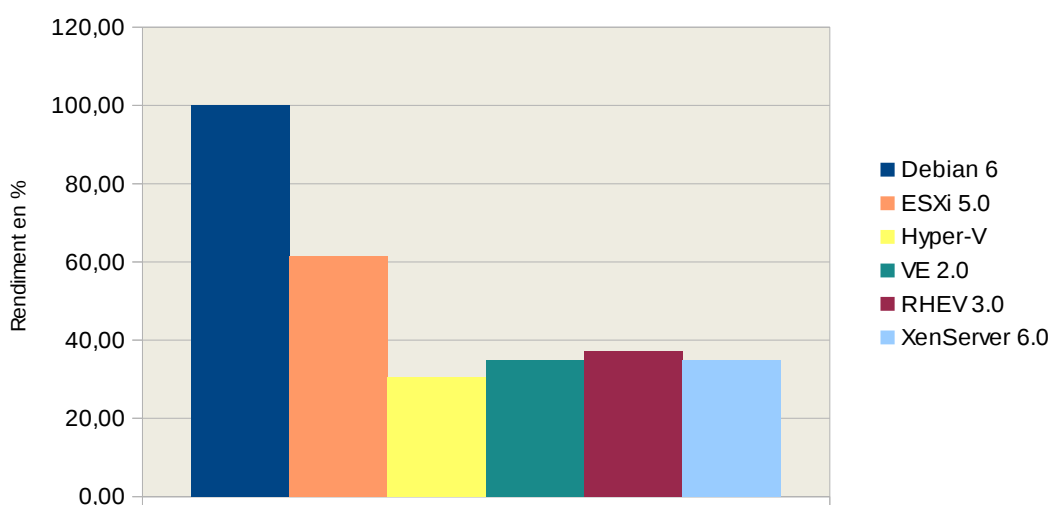
Il·lustració QQ: Benchmark : Disc - Postmark

7.7.5. Mitjana dels resultats

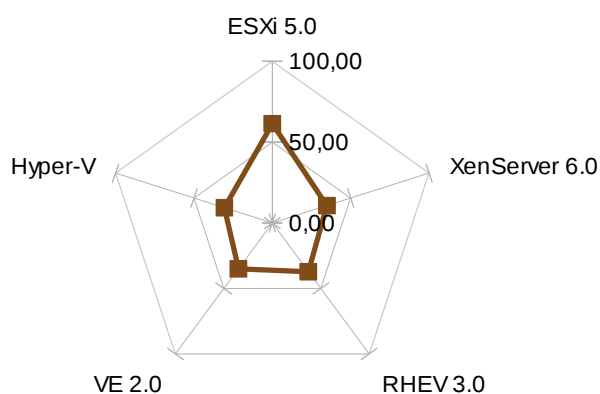
Categoria **Disc** Test **Mitja**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Mitja	1737,00	399,00	534,00	520,00	551,00	592,00
%	100,00	61,49	30,55	34,98	37,25	34,85

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració RR: Benchmark - Mitja discos

7.7.6. Resultats obtinguts

FS-Mark

Veim dos resultats totalment oposats. Quant al test de transferir dades, aquest valor és millor quan menor sigui. Podem veure com les solucions de VMware, Proxmox i Xenserver són les millors en aquest tipus d'accés a dades, amb "No sync/Fsync".

El rendiment dels disc durs és simplement molt inferior a l'esperat, no prevèiem que el rendiment dels discos DDS virtualitzats fos tan baix en comparació amb la màquina real.

En aquest cas parlem de percentatges; per tant, entenem que, si els discos haguessin estat molt més ràpids, el percentatge hauria estat el mateix.

Podem veure que, en aquests casos, les solucions perden molt de temps, quasi el doble a l'hora de realitzar operacions de disc. En aquest cas sí que la capa de virtualització penalitza molt el rendiment, i haurem de tenir en compte quin tipus d'aplicacions volem instal·lar sobre un sistema virtualitzat per determinar si ens interessa mantenir les dades a un disc visualitzat o bé a una partició no virtualitzada, per a millorar el rendiment.

També interessarà saber el rendiment del sistema i ús que se li donarà, per seleccionar un tipus d'emmagatzemament o un altre.

7.8. Sistema

7.8.1. PHPBench

Descripció: PHPBench és una *suite* de referència per a PHP.

Es realitza un gran nombre de proves senzilles per tal de comprovar diverses característiques de l'interpret de PHP.

PHPBench es pot utilitzar per comparar maquinari, sistemes operatius, versions de PHP, els acceleradors i memòria cau (*cache*) del PHP, les opcions del compilador, etc.

7.8.2. Gràfiques

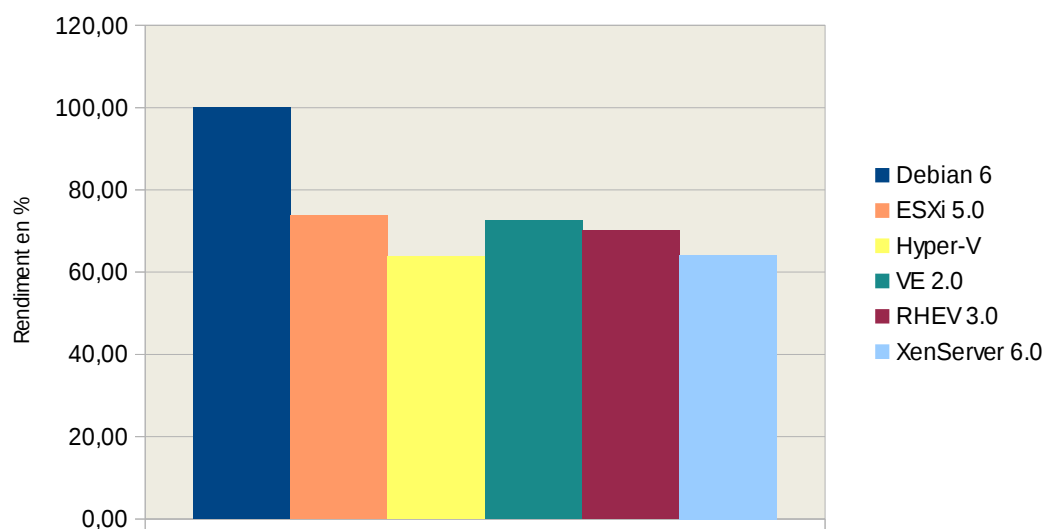
Categoria **Disc**

Test **Mitja**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Mitja	1737,00	399,00	534,00	520,00	551,00	592,00
%	100,00	73,86	63,85	72,55	70,16	64,18

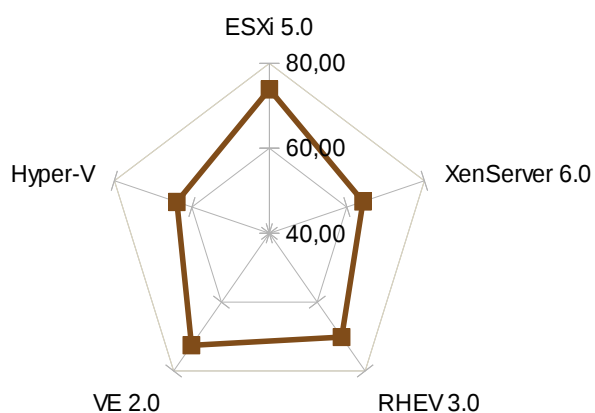
Representació gràfica

Barres



Representació gràfica

Xarxa



Il·lustració SS: Benchmark - Sistema Phpbench

7.8.3. Apache

Descripció: és un test per avaluar el rendiment del servidor web Apache.

L'objectiu és veure quantes respostes per segon pot sostenir, el simulacre que es realitza és el de rebre el servidor web 700.000 sol·licituds d'entrada i 100 sol·licituds simultànies de sortida.

Dependències de programari:

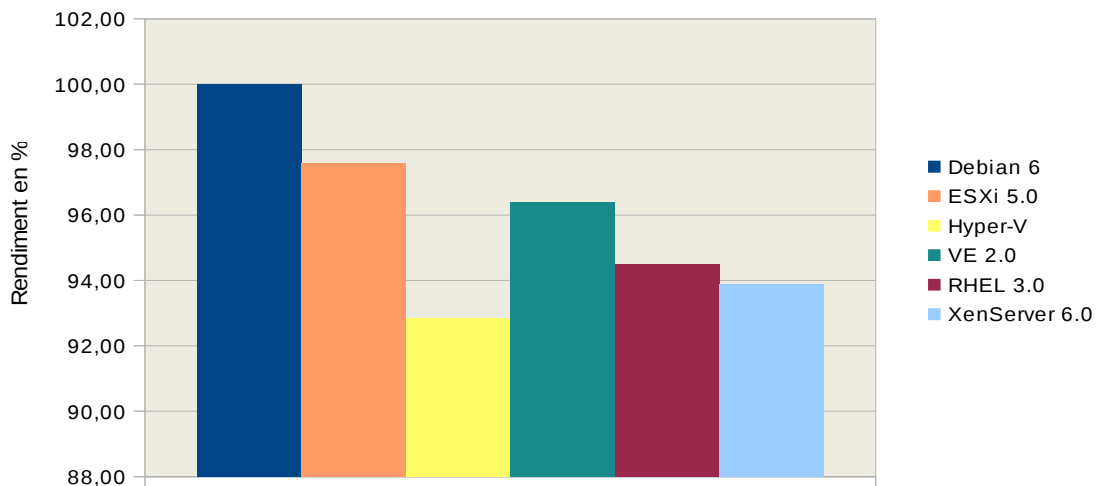
- Compiler / Development Libraries

7.8.4. Gràfiques

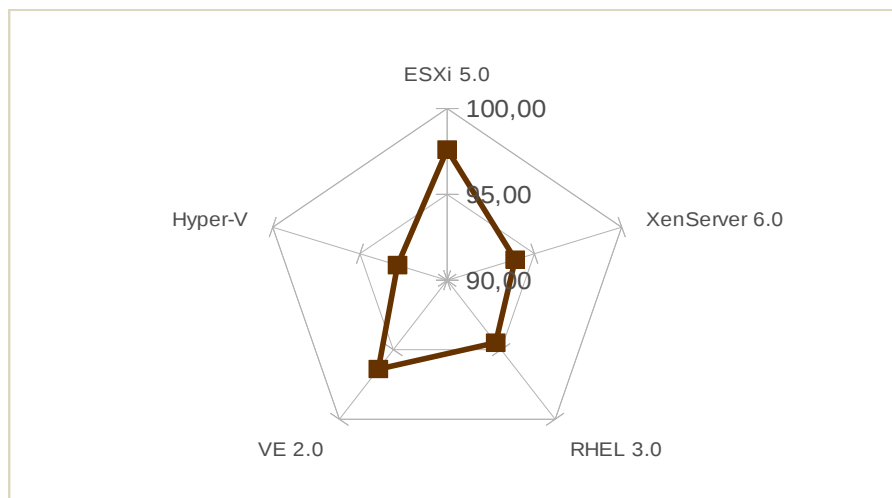
Categoria **Sistema** Test **Phpbench**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEL 3.0	Citrix XenServer 6.0
Segons	47066	45935	43700	45365	44473	44189
%	100,00	97,60	92,85	96,39	94,49	93,89

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració TT: Benchmark Sistema - Apache

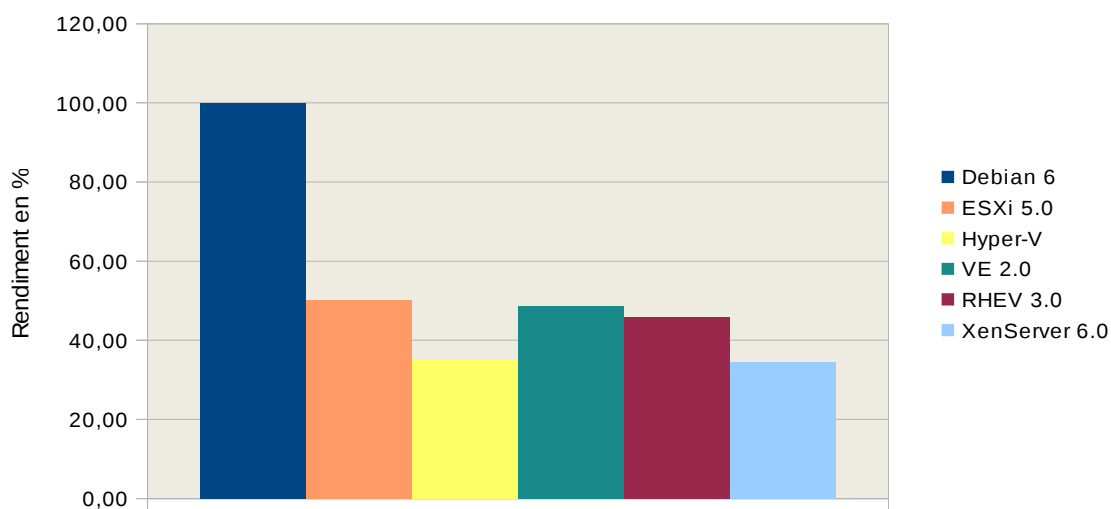
7.8.5. Mitjana

Categoria **Sistema**

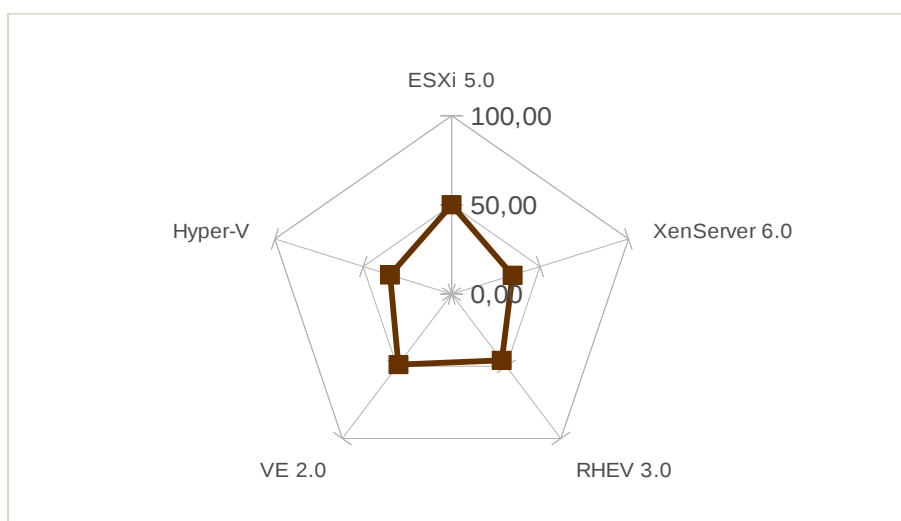
Test **Apache**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
Segons %	7984,93	4002,72	2783,24	3889,87	3659,76	2753,05
%	100,00	50,13	34,86	48,72	45,83	34,48

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració UU: Benchmark Sistema Mitja

7.8.6. Resultats obtinguts

PHPBench

El rendiment percentual obtingut sobre aquestes proves indica gran nivell de rendiment en les màquines utilitzades; no obstant això, es destaquen les solucions de VMware i Proxmox, la solució de Microsoft amb el seu Hyper-v queda la darrera. Així i tot, el seu rendiment se situa prop del 93 % en comparació amb la màquina original.

Millor	Pitjor
RHEV	XenServer
Hyper-V	Proxmox

Amb "No sync/Fsync" els millors.

Millor	Pitjor
VMware	Hyper-V
Proxmox	RHEV

Veim clarament una inversió de l'ordre quant als resultats obtinguts. Així doncs, entenem que cada tecnologia empra un tipus d'accés al disc totalment oposat a l'altre.

Apache

El test de rendiment del servidor Apache en línies generals ha estat bastant fluix, entenem que el test fa bastant d'ús del sistema de disc i, com ja hem vist al rendiment de disc, és el problema més greu que tenen aquestes solucions.

No obstant això, podem destacar VmWare seguit molt de prop de Proxmox com les millors de les solucions que ha superat el test, i la de Microsoft i Citrix com a pitjors resultats obtinguts.

En aquest cas en concret podem veure com les solucions que empen Xen i la tecnologia pròpia de Microsoft són les de pitjor resultat, per contra VMware i les dues solucions basades en KVM són les millors.

7.9. Xarxa

Per poder fer un test de la xarxa, el PTS només ens presentava un test anomenat Network-Loopback. Aquest en els màquines que vàrem provar retornava un error d'execució, i vàrem optar per desenvolupar nosaltres mateixos un test.

El test consistia a connectar-se a un directori compartit a una altra màquina de la xarxa i descarregar una sèrie de fitxers de distinta grandària; n'enregistràrem els temps obtinguts, tant de descàrrega com de pujada d'aquests.

Com que enteníem que en aquests tests hi ha molts de factors externs als servidors en si que estàvem avaluant, es va fer que el traspàs de dades es realitzàs directament d'ordinador a ordinador mitjançant un cable de xarxa RJ-cat5 cross-over. D'aquesta manera evitàvem possibles interferències de factors externs com ara altre tràfic no contemplat de la xarxa o fins i tot interferències que puguin ocasionar l'electrònica de xarxa, *switch*, *routers*...

Així, partint d'un mateix marc podem veure com és el rendiment dels sistemes emprant la xarxa.

7.9.1. Script test xarxa

Per realitzar les proves abans esmentades, un cop connectats els ordinadors, es va procedir a compartir una carpeta a un ordinador auxiliar i a copiar-hi a dins un fitxer de 648 MB. Per a més informació direm que es tractava d'una ISO de la Debian Testing 6.0.

L'script consisteix en 3 instruccions:

La primera és l'encarregada de fer la connexió entre la màquina virtual i l'ordinador que fa de servidor de fitxers Samba. Ens munta la unitat com si d'una unitat local es tractàs.

A continuació les dues línies següents s'encarreguen respectivament de descarregar el fitxer del servidor Samba cap a la nostra màquina virtual i enregistrar el temps que ha tardat, mitjançant el comandament "time".

Com a inicis direm que el comandament "time" abans de qualsevol instrucció i en la mateixa línia fa retornar-nos el temps que ha tardat a executar-se la instrucció que el precedeix.

La segona línia fa el procés a la inversa, puja el fitxer de tornada a una altra carpeta al servidor Samba.

```
#muntar unitat
smbmount //192.168.0.211/public /media/cdrom -o username=root,password=2443
```

```
#test de baixada
(time cp /media/cdrom/down/debian.iso ~/) 2> down.txt

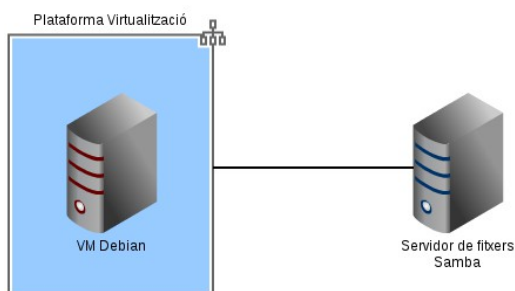
#test de pujada
(time cp debian.iso /media/cdrom/up/) 2> up.txt
```

A l'hora d'avaluar el rendiment de la xarxa ens trobarem amb un gran impediment per part de la *suite* PTS: no hi havia test específic per a xarxa; millor dit, n'hi havia un, però no funcionava. El test en si anomenat Network-Loopback teòricament avaluava el rendiment del Loopback de les màquines, però com hem comentat quan l'executàrem no acabava correctament. Aleshores procedirem a realitzar nosaltres mateixos un test de xarxa.

Esquema

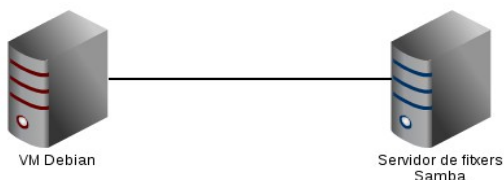
El test que realitzaríem consistiria a connectar 2 ordinadors: un faria de servidors de fitxers Samba i l'altre seria la plataforma virtualitzadora amb la màquina virtualitzada allotjada a l'interior.

L'esquema físic seria el següent:



Il·lustració VV: connexió plataforma virtualització - servidor fitxers

L'esquema lògic seria aquest altre:



Il·lustració WW: connexió VM Debian - servidor fitxers

Un cop que tenguérem la connexió física establerta, realitzarem les següents operacions, scripts per procedir amb el test.

Crearem una sèrie de fitxers que emprarem per transferir entre els ordinadors.

7.9.2. Generació de fitxers – crear.sh

En aquest punt ens trobarem que necessitàvem una sèrie de fitxers i dades per poder transmetre d'un ordinador a l'altre per poder avaluar el rendiment de la xarxa. Així doncs, dissenyarem el següent script per crear tants de fitxers com creguéssim necessaris.

Atès que genera fitxers, férem 5 grups, cada un d'ells de 200 MB de grandària:

- 200 fitxers d'1 MB
- 20 fitxers de 10 MB
- 2 fitxers de 100 MB
- 1 fitxer de 200 MB

```
#!/bin/bash
for i in `seq 1 $1`; do
    echo $i
    dd if=/dev/zero of=./f$2MB$i bs=$2MB count=1
done
```

La cridada es féu de la següent manera:

```
sh crear.sh 2 100
```

El primer paràmetre és el nombre de fitxers a crear i el segon, la grandària dels fitxers expressats en megabytes.

A l'exemple anterior, ens generaria dos fitxers de cent megabytes cada un amb els noms f100B1 f100MB2. Ens indiquen que són, com hem dit abans, fitxers de 100 MB i la numeració que han adquirit a l'hora de la creació.

7.9.3. Còpies

Un cop fet, realitzarem la còpia de cada carpeta de l'origen al destí i de viatge de tornada per poder calcular la velocitat de baixada i pujada del sistema.

Per fer això crearem aquest script:

```
#montar unitat
smbmount //192.168.0.211/public /media/cdrom -o username=root,password=2443

#test de baixada
(time cp /media/cdrom/Informatica/iso/down/Hyperv.iso ~/) 2> down.txt
```

```
#test de pujada
(time cp Hyperv.iso /media/cdrom/Informatica/iso/up/) 2> up.txt
```

7.9.4. Problemàtica

A causa dels resultats obtinguts, vàrem veure que el rendiment de xarxa era molt pobre, com podem veure a l'estadística aquí publicada. Gràcies que ja teníem calculats els tests dels rendiments dels disc durs, vàrem tenir la sospita que el baix rendiment de la xarxa no era degut a si mateixa, sinó al coll de botella que eren els discs durs.

Per tal de validar aquesta teoria i per eliminar el factor dels discs durs, tenguérem la idea de crear discs durs a la RAM dels ordinadors. D'aquesta manera, com que la memòria és infinitament més ràpida que els discs durs i que la mateixa xarxa, eliminaríem un factor que *a priori* enteníem que ens desvirtuava els resultats. Així doncs, seguïrem aquestes passes.

Generació de fitxers

En aquest exemple veim com feim un bucle de 1.000 iteracions a on a dins es crea un fitxer d'1 MB de grandària. Canviant aquests 2 valors podem fer tants de fitxers i de les grandàries que vulguem com siguin necessaris.

```
#!/bin/bash
for i in `seq 1 1000`; do
    echo $i
    dd if=/dev/zero of=./f1MB$i bs=1MB count=1
done
```

Creació de discos RAM

Aquí podem veure amb aquest script que ens permet la creació d'un disc RAM de 250 MB.

crearRAM.sh

```
#!/bin/bash
#montar ramdisk i montar
mount -t tmpfs none /home/tomeu/ramdisk -o size=250m
```

L'script s'executa tant en l'ordinador *host* com el client.

Execució del *benchmark*

A l'ordinador client executarem aquests scripts un cop creat el disc RAM.

```
#!/bin/bash
# copia fitxers 1M
echo PREPARANT FITXERS DE HD A RAMDIKS

# agafa data inicial
res1=$(date +%s.%N)

rm -r ./ramdiskr/*          # esborra continguts de discs RAM
remot
rm -r ./ramdisk/*          # esborra continguts de discs RAM
local

cp -R ./public/down/$1/ ./ramdiskr/  # copia el directori on tenim els
                                       # fitxers
                                       # creats al disc RAM
                                       #test de baixada

echo iniciant el test de baixada
res1=$(date +%s.%N)        # captura temps inicial

for i in `seq 1 5`; do     # bucle de copia inicialment 5 cops
    cp -R ./ramdiskr/ ./ramdisk/
done

res2=$(date +%s.%N)       # captura de temps final

echo "Temps "$1" : $(echo "$res2 - $res1"|bc )" > down$1.txt
                           # guardem el resultat
                           # del temps de baixada
                           # test de pujada

rm -r ./ramdiskr/*        # esborrem contingut a disc Remot
echo iniciant test de pujada

res1=$(date +%s.%N)      # captura de data
for i in `seq 1 5`; do   # bucle de copia inicialment 5 cops
    cp -R ramdisk/* ramdiskr/  # copia de RAM local cap a server
done
res2=$(date +%s.%N)      # captura data final proces
echo "Temps "$1" :    $(echo "$res2 - $res1"|bc )" > up$1.txt
                           # guardar resultat

echo test finalitzat
```

Un cop que tenim els discos RAM als dos servidors, els fitxers que emprarem per copiar d'un a l'altre, l'script fa és el següent (recordem que per simplificar el procés s'executa tot des del client, és a dir la màquina virtual):

1. Ens connectam als dos directoris compartits del servidor Samba, un és un directori on tenim els fitxers creats per a tal propòsit i l'altre és el disc RAM.
2. Es copien els fitxers que estan emmagatzemats al disc dur cap al disc RAM del servidor Samba.

3. Quan els fitxers estan al disc RAM, descarregam aquests des del servidor cap a la màquina virtual.
4. Es computa el temps que s'ha tardat a fer l'operació.
5. Els mateixos fitxers, que ara es troben al disc RAM de la màquina virtual, s'apugen cap al disc RAM del servidor Samba.
6. Es computa el temps que s'ha tardat.

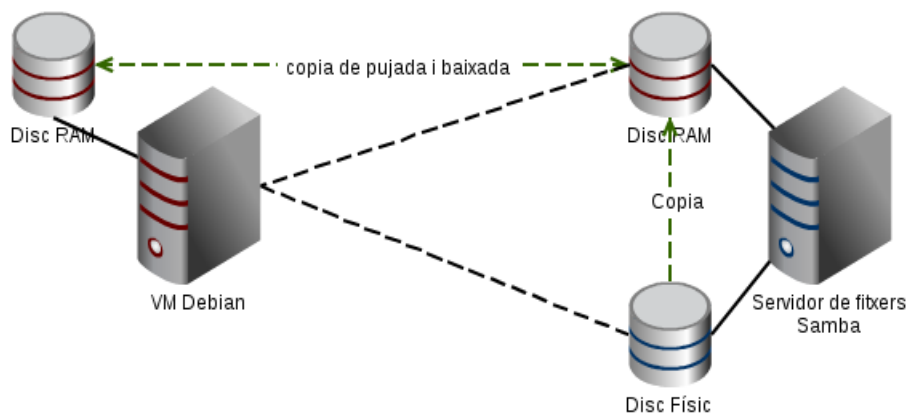
Així anam repetint tantes vegades com vulguem l'script i tants de tipus de fitxers com vulguem.

Per a aquest cas hem creat un altre script, que allò que fa és cridar el que ho executat tot i li envia els paràmetres desitjats.

```
Sh 3MainCopia.sh
```

```
#!/bin/bash
sh 3copia.sh 1MB
sh 3copia.sh 10MB
sh 3copia.sh 100MB
sh 3copia.sh 1000MB
```

Com podem veure l'script executa 4 vegades l'script de còpia amb els paràmetres de copiar fitxers de diferent grandària.



7.9.5. Gràfiques

7.9.5.1. UPLOAD

Categoria **Xarxa** **Test Upload**

Valors reals

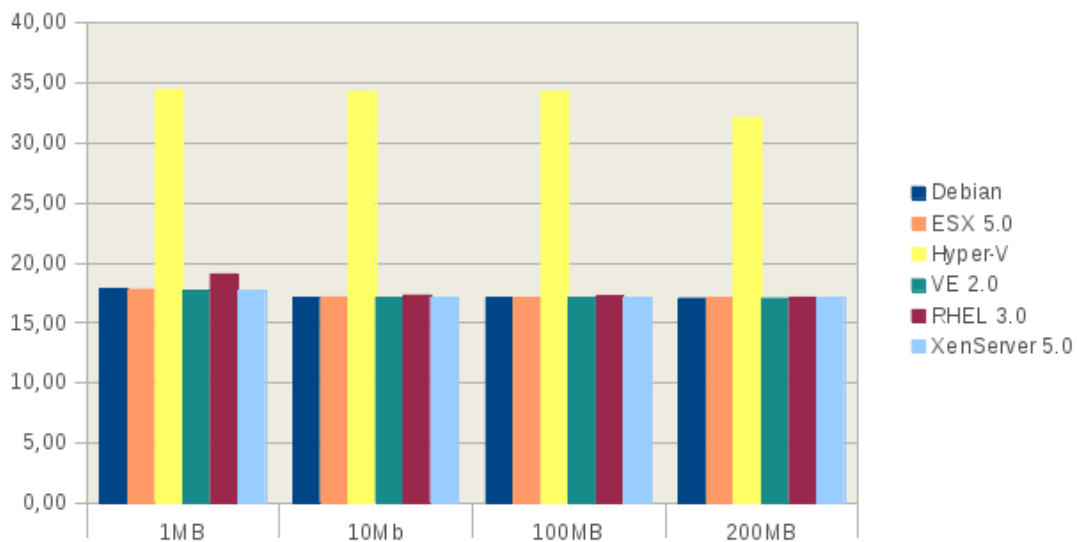
	Community Debian	VmWare ESX 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEL 3.0	Citrix XenServer 5.0
1MB	17,86	17,80	34,44	17,69	19,07	17,66
10Mb	17,11	17,16	34,19	17,13	17,31	17,14
100MB	17,06	17,13	34,30	17,06	17,26	17,13
200MB	17,06	17,08	32,11	17,06	17,15	17,08
Mitja	17,27	17,29	33,76	17,24	17,70	17,25

Valors percentuals %

	Community Debian	VmWare ESX 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEL 3.0	Citrix XenServer 5.0
1MB	100,00	100,00	51,85	100,00	93,63	100,00
10Mb	100,00	99,74	50,04	99,88	98,85	99,84
100MB	100,00	99,61	49,75	100,00	98,87	99,63
200MB	100,00	99,88	53,12	100,00	99,50	99,89
Mitja %	100,00	99,81	51,19	99,97	97,71	99,84

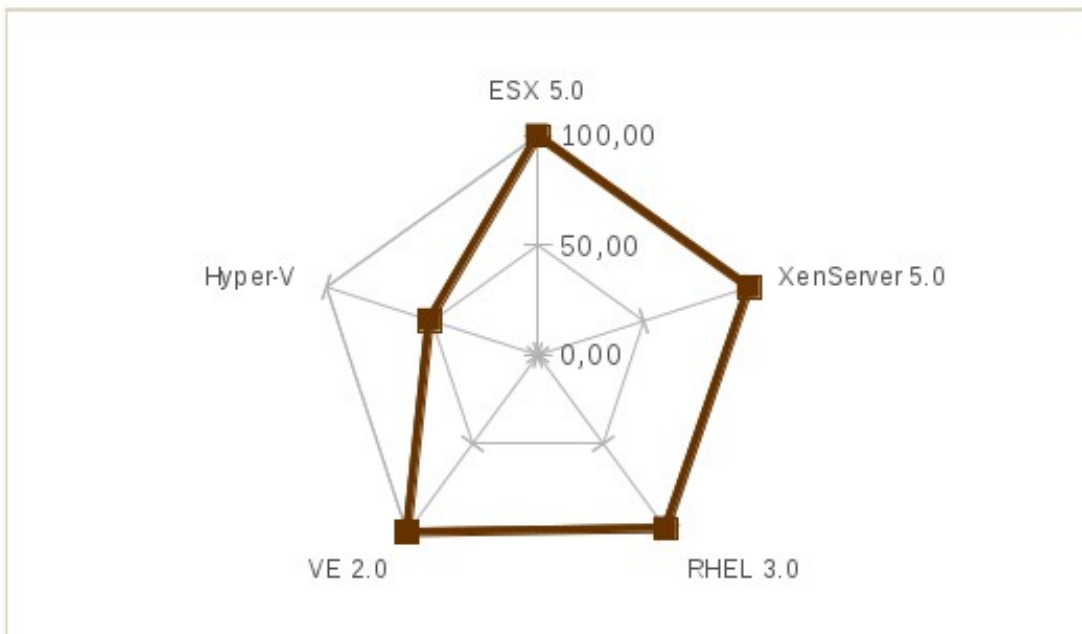
Il·lustració XX: Benchmark Xarxa Upload

Representació gràfica **Barres**



Il·lustració YY: Benchmark barres - Xarxa Upload

Representació gràfica **Xarxa**



Il·lustració ZZ: Benchmark xarxa - Xarxa Upload

7.9.5.2. DOWNLOAD

Categoria **Xarxa**

Test Download

Valors reals

	Community Debian	VmWare ESX 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEL 3.0	Citrix XenServer 5.0
1MB	17,88	18,04	26,86	17,97	20,48	17,89
10Mb	17,51	17,17	22,66	17,24	18,08	17,18
100MB	17,39	17,15	23,15	17,09	17,76	17,17
200MB	17,43	17,08	22,50	17,08	17,56	17,09
Miija	17,56	17,36	23,79	17,35	18,47	17,33

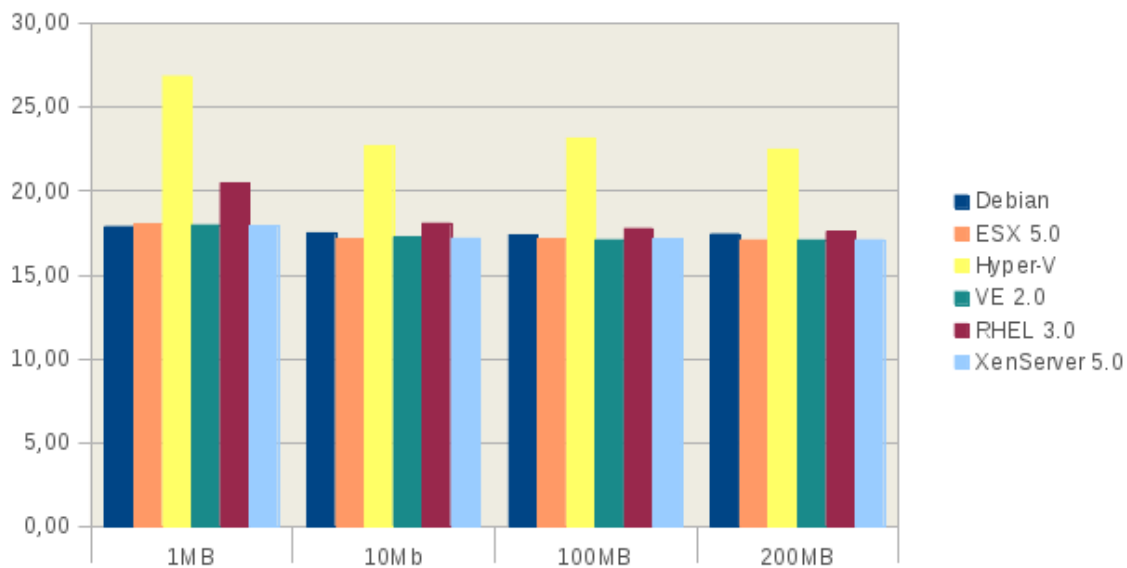
Valors percentuals %

	Community Debian	VmWare ESX 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEL 3.0	Citrix XenServer 5.0
1MB	100,00	99,12	66,58	99,50	87,31	99,94
10Mb	100,00	100,00	77,28	100,00	96,87	100,00
100MB	100,00	100,00	75,12	100,00	97,93	100,00
200MB	100,00	100,00	77,49	100,00	99,24	100,00
Miija %	100,00	99,78	74,12	99,88	95,34	99,98

Il·lustració AAA: Benchmark Tabla Valors Test xarxa dowload

Representació gràfica

Barres



Il·lustració BBB: Benchmark xarxa download, grafica de barres

Representació gràfica

Xarxa



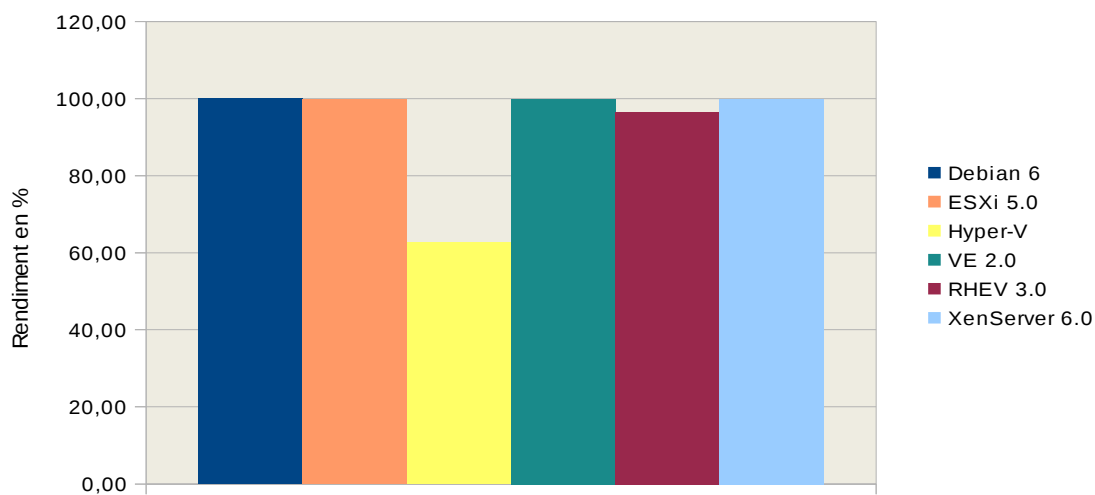
Il·lustració CCC: Benchmark xarxa download, grafica de xarxa

7.9.5.3. Mitjana

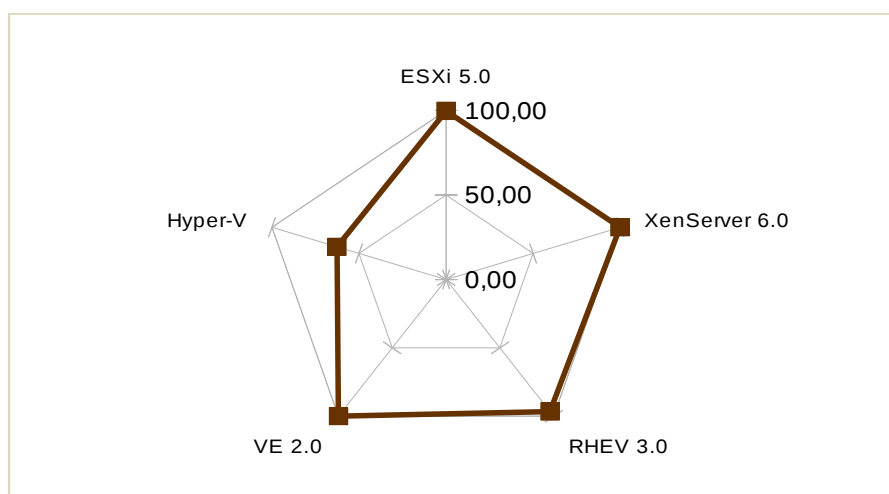
Categoria **Processador** Test **Mitja**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
%	100	99,79	62,66	99,92	96,52	99,91

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració DDD: Benchmark - Xarxa Mitja

7.9.6. Resultats obtinguts

La gran diferència de velocitat entre la transferència de dades d'un disc dur (HD) i un sistema ràpid com ara la RAM fa que el primer s'hagi constituït des de fa molt de temps al món de la informàtica en un dels colls de botella més coneguts per a qualsevol sistema informàtic. De fet, actualment encara és el dispositiu més lent intern de què disposa un ordinador. Per tant, era de suposar que el pitjor rendiment en la prova dels sistemes seria en aquest aspecte, i aquest és tan gran que afectava directament el rendiment de la transferència de xarxa emmascarant el seu resultat.

Així, per evitar aquesta deformació en el resultat, es crearen discs durs RAM, o també anomenats virtuals, a la memòria dels ordinadors. D'aquesta manera, hem pogut desfer-nos del problema de la lentitud de transferència dels discos durs actuals i centrar-nos en allò que seria la velocitat de transferència pura de la xarxa.

Aleshores podem veure com la transferència de xarxa és quasi la mateixa que la màquina que empram com a model i veure que és una de les millors implementacions realitzades. A més, tenint en compte que parlem de servidors, està clar que és molt important aquest rendiment.

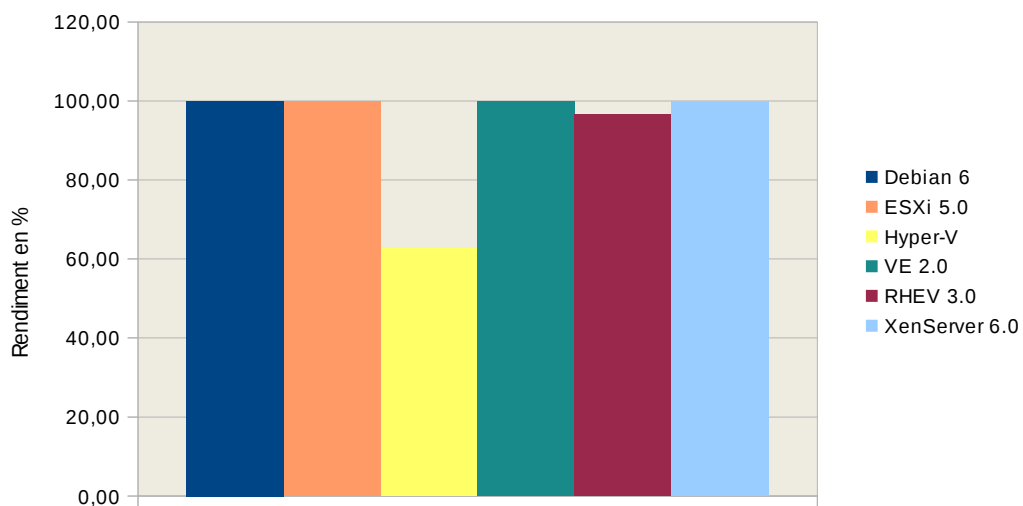
El cas excepcional en aquest test és l'HyperV, suposam que és atès que aquest sistema disposava d'un *driver* "egacy", una espècie de *driver* per sortir del pas fins que se n'instal·lès un d'adient per a la interacció entre la màquina virtualitzada i l'hipervisor, pel fet de no disposar de *driver* específic per a aquest sistema; la prova va ser molt penalitzada en aquest cas. Ara bé, suposam que l'ús d'un *driver* desenvolupat per la màquina de proves específica implicaria un rendiment similar a les altres solucions.

7.10. Resum estadístic general

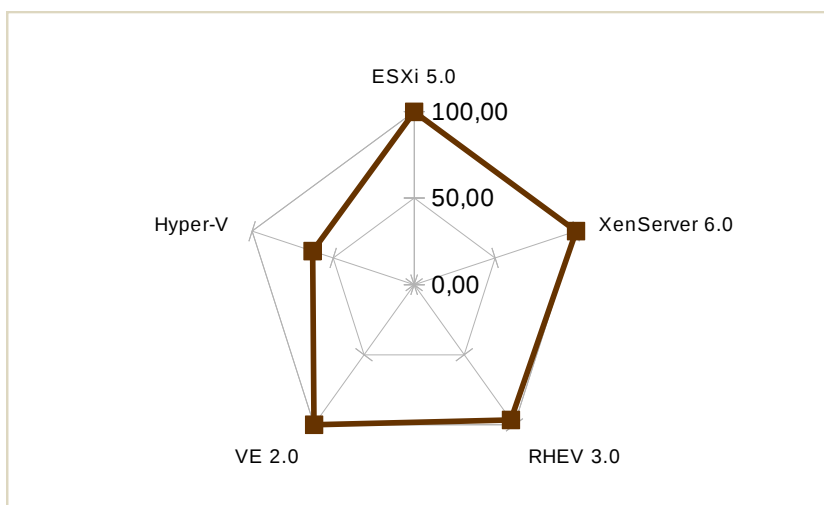
Categoria **Processador** Test **Mitja**

	Community Debian 6	VmWare ESXi 5.0	Microsoft Hyper-V	Proxmox VE 2.0	Red Hat RHEV 3.0	Citrix XenServer 6.0
%	100	99,79	62,66	99,92	96,52	99,91

Representació gràfica **Barres**



Representació gràfica **Xarxa**



Il·lustració EEE: Benchmark - Visió global

7.10.1. Resultat finals. Conclusions

Podem veure a l'apartat anterior una mitjana de totes les proves realitzades i plasmades sobre una gràfica en forma de xarxa, la qual ens fa d'una forma molt fàcil veure els resultats de les solucions i en quin estat es troba cada solució.

Hi podem veure dues gràfiques: una que ho representa per solucions i l'altra, per serveis.

Gràfica de serveis / recursos de servidors

En aquesta gràfica, hi podem observar de forma clara que els millors recursos dels sistemes de virtualització són per aquest ordre: memòria, xarxa, CPU, disc i sistema.

Què podem dir d'aquesta gràfica i resultats?

Per una banda, són uns resultats una mica esperats, atès que els dispositius que són més ràpids han demostrat un rendiment més proper a un entorn real; per contra, els dispositius més lents han exagerat els resultats de forma negativa.

Apreciam també lligat al punt anterior que, a mesura que un recurs és de cada cop més lent, el sistema de virtualització agreuja notablement la mancança del dispositiu. És a dir, la memòria RAM és el dispositiu més ràpid que podem emprar; per tant, els sistemes de virtualització surten menys penalitzats en aquest camp. Per contra, les lectures i escriptures al disc dur són l'apartat més penalitzat de tots, ja de per si és el dispositiu més lent avui dia, quan es virtualitza l'agreujament de la situació és molt destacable.

Entenem que a mesura que vagin avançant quant a velocitat els mitjans d'emmagatzemament serà un fort impuls per als sistemes de virtualització; així mateix, els nous discs durs de tecnologia SSD seran una gran ajuda per a aquestes plataformes.

http://en.wikipedia.org/wiki/Solid-state_drive

No obstant això, entenem que a nivell percentual la pèrdua serà la mateixa si repetim aquests tests sobre discos SSD, no en canvi el seu rendiment i temps de lectures i escriptures, que de ben segur milloraria en una proporció de com a mínim el doble, ateses les característiques d'aquests nous suports.

Solucions de virtualització

Per altra banda, també manipulant les dades, podem veure a la gràfica que representa els *benchmarks* quins tests van millor i pitjor en global en el món de la virtualització.

De les dades extretes dels tests, se'ns fa una mica difícil poder dir quina és la millor solució i quina és al pitjor que hi ha actualment al món de la virtualització. En ocasions tenim solucions que destaquen un poc per damunt d'altres en algun aspecte en concret, però tal vegada després penalitzen en altres aspectes.

Del que sí que podem parlar sense equivocar-nos és d'un parell de punts:

- VMware tal volta segueix sent la solució més completa, atesa l'experiència que té des de fa molts d'anys en el món de la virtualització, però les altres solucions quant a rendiment li estan agafant terreny. El seu client, no obstant això, ara per ara és el més complet de tots.
- XenServer és una solució que sense fer gaire renou i amb molta velocitat s'està introduint al mercat, els seus rendiments són molt bons i fins i tot en alguns punts ha destacat notablement.
- Red Hat basta sobre el nucli de Red Hat i programari lliure, es desenvolupa molt bé contra la competència, de ben segur serà seleccionat per empreses i entitats que ja estiguin treballant amb aquesta empresa. El seu rendiment és més que acceptable, però haurà de reescriure el codi del seu client perquè pugui ser emprant per altres plataformes i no sigui tan primmirat en els requisits de funcionament.
- Quant a l'Hypervisor, de Microsoft, en aquesta prova hem vist el gran problema que suposa no disposar dels *drivers* adients per a la majoria de sistemes operatius del mercat. La plataforma és totalment diferent a totes les emprades, ja que és l'única que no empra res relacionat amb el programari lliure ni amb cap nucli de Linux.
- Proxmox és l'antagonisme de l'Hypervisor, una eina basada en el 100 % del programari lliure, amb un bon rendiment i una cosa que no tenen els altres: la subjecció a llicències. La versió més bona de Proxmox és la que hem provat, és a dir, hi ha una versió i res més, però aquesta disposa de moltes opcions de les quals les altres només disposen en versions més avançades i de pagament, com ara l'alta disponibilitat (HA). Això sí, al test de CPU va donar un rendiment inferior als altres, especialment penalitzat en el test de gZip. Costa una mica d'entendre'n el motiu, tal vegada el fet de dur el Kernel més modern que els altres hi tengués a veure; serà un punt a tenir en compte.

8 Conclusions

El treball s'ha centrat en 5 solucions avui dia les més reconegudes en el panorama de la virtualització.

Hi ha coses que ens han sorprès i d'altres que no donat els resultats obtinguts.

Vmware segueix sent la eina més homogènia en tots els camps, n'hi ha que en alguns llocs li superen, però en altres punts empitjoren molt

El que més ens ha sorprès és el baix rendiment en quan a accés al sistema d'emmagatzemament dels discs dur, està clar que s'esperava un rendiment inferior a un disc no virtualitzat, però no que aquests dones un rendiment tan baix.

Gran implantació del programari lliure en aquestes solucions, quatre de les cinc testejades, estan basades parcialment o totalment sobre programari lliure, llevat de la solució de microsoft que empra tecnologia 100% seva.

Les solucions d'empreses consolidades no son millors que les basades en programari lliure, ni les basades en programari lliure donen un rendiment baix.

9 Línies de Millora

A mida que anàvem realitzant el projecte, varem veure una sèrie de punts en els quals ens interessaria poder seguir investigant, provant, experimentant i anar seguint extreiem nous tests sobre aquests tipus de tecnologies. Com a millores del projecte o línies de futur establiríem les següents :

9.1 Proxmox, proves de HA

Sabent que la solució Proxmox VE, permet realitzar alta disponibilitat (HA), seria molt interessant el poder realitzar una serie de proves sobre aquesta funció. Consistiria en emprar com a mínim dues màquines que fecin les funcions de nodes HA i una tercera que fes les funcions de cabina SAN a on allotjar les màquines virtuals. A les hores, un cop muntat tot, desconectar el node que substena una màquina virtual i veure com el segon node agafa el control de la màquina.

9.2 Realitzar més proves de CPU

Atès que les dues proves de rendiment de CPU han donat uns rendiment molt extrems en cada una d'elles, seria interessant realitzar més proves sobre el ús exhaustiu de CPU. Emprant per això altres tests del Phoronix Test Suite o emprar-ne un altre com ara el Tsung Test tsung.erlang-projects.org/

9.3 Test de disc sobre altres tecnologies

Donat al baix rendiment recollit de les transferències dels disc durs i veient que a menor velocitat del dispositiu, el rendiment de la màquina virtual empitjora de forma incremental.

Estaria molt be el poder realitzar les mateixes proves d'accés sobre els disc durs, però emprant altres tipus de tecnologies o una combinació d'aquestes, com ara realitzar les proves sobre discos amb interfície SCSI, SATA o SAS. També es podria fer la combinació de les tecnologies abans comentades amb diferents tipus de discos, com ara els disc durs convencionals, hard disk drive (HDD) o els actuals solid-state drive (SSD) que tenen un rendiment molt elevat.

9.4 Unitat de dades no virtualitzada

Un test que seria interessant realitzar seria el testear la transferència de dades en una unitat d'emmagatzemament virtualitzada, contra el mateix test i màquina virtual, però que accedeixi a una unitat de disc no virtualitzada. Es a dir, tenir un sistema híbrid, servidor virtualitzat i unitat d'emmagatzemament no virtualitzat.

9.5 Driver Hyper-V

Seria interessant repetir els tests de la solució de Microsoft un cop disposéssim del driver de xarxa correcte per a la màquina virtual que hem emprat per fer les proves.

10 Bibliografia i Referències consultades

10.1 Història de la virtualització

VmWare	vmware.com/es/virtualization/virtualization-basics/history.html
Blog ConsultaUnitPro	consultaunitpro.com/tag/historia-de-la-virtualizacion
History of virtualization	en.wikipedia.org/wiki/Timeline_of_virtualization_development

10.2 Tipus de virtualització

Blog vmlogia	vmlogia.com/tiposdev.aspx
Wikipedia (EN)	en.wikipedia.org/wiki/Virtualization
Kernelthread	kernelthread.com/publications/virtualization/
KVM	linux-kvm.org/page/Main_Page
QEMU	wiki.qemu.org/Main_Page
OpenVZ	wiki.openvz.org/Main_Page

10.3 Programari de Virtualització

Vmware ESXi 5.0	vmware.com
Proxmox VE 2.0	proxmox.com
Citrix Xen Server 6.0	citrix.com
Red Hat Enterprise Virtualization 3.0	redhat.com
Hyper-V microsoft	microsoft.com

(*) no posem els enllaços directes al programari, degut al constant desenvolupament dels productes, fa que els links es quedin antiquats en qüestió de mesos.

10.4 Eines de Benchmark

Samba	samba.org
Phoronix Test Suite	phoronix-test-suite.com