Treball fi de carrera

ENGINYERIA TÈCNICA EN INFORMÀTICA DE SISTEMES

Facultat de Matemàtiques
Universitat de Barcelona

APROFITAMENT DE RECURSOS COMPUTACIONALS ESTÀNDARD PER A LA CREACIÓ D'UN CLÚSTER DE CÀLCUL

Jordi José Bazán

Director: Jaume Timoneda Salat
Realitzat a: Departament de Matemàtica Aplicada i Anàlisi. UB

Barcelona, 12 de juny de 2012

Contingut

Aprofitament de Recursos Computacionals Estàndard per a la Creació d'un Clúster de Càlcul....5
Introducció als Sistemes de Càlcul d'Altes Prestacions....6
Orígens....6
Tipus de Càlcul d'altes prestacions....6
Computació paral·lela....6
Computació Distribuïda....7
Sistemes de Càlcul d'altes prestacions a l'estat Espanyol....8
Centro de Supercomputación de Galicia (CESGA)....8
Elements Clau en Sistema HPC....10
Elements de Maquinari....10
Elements de Càlcul....10
CPU....10
GP/GPU....12
Memòria RAM....13
NIC i NPU....14
Scratch....15
Elements de Xarxa....15
Xarxa de Càlcul....17
Xarxa de Dades....17
Xarxa d'Accés....17
Elements d'Emmagatzematge....18
Sistemes de fitxers distribuïts d'alt rendiment....18
Elements de Programari....20
Sistemes Operatius....20
Gestors de cues....21
Parel·lització....22
Compiladors....23
Programari de gestió i control....24
Consideracions Tècniques....25
Infraestructura de Servidors....25
Maquinari....25
Programari....25
Infraestructura d'Aula....27

Maquinari 27
Programari 27
Paravirtualització amb XEN 27
Màquines Virtuals 28
Gestor de Cues 31
Grid Engine Project 31
Open Grid Scheduler 32
Rols dels Nodes 32
Funcionament del sistema de cues 34
Scheduler 36
Comandes Bàsiques 37
Configuració Bàsica 38
Configuracions complexes OGS 40
Muntatge de Xarxa 42
Virtual Network Interface (VIF) 43
Virtualització xarxa amb XEN 43
Integració amb la Xarxa i Serveis de la Facultat 44
Serveis LDAP i d'Usuari 44
Serveis Gestió d'Usuari 44
Proposta Tècnica 45
Introducció 45
Infraestructura de Servidors 46
Maquinari 46
Programari 46
Estructura Gestor Cues dins de l'entorn del projecte 47
Estructura d'usuari i accés 47
Infraestructura d'Aula 48
Muntatge de Xarxa 48
Muntatge Final 48
Muntatge Clients 50
Anàlisi de Costos 50
Proposta ideal 51
Introducció 51
Infraestructura de Servidors 52

Maquinari 52
Programari 53
Granja de Servidors Virtuals 53
Estructura Gestor Cues dins de l'entorn del projecte 54
Muntatge de Xarxa 54
Infraestructura d'Aula 55
Muntatge de Xarxa 55
Muntatge Clients 55
Anàlisi de Costos 55
Solucions Intermèdies 57
Introducció 57
Anàlisi de Costos 58
Perquè no somiar? 60
ANNEX I – INSTAL·LACIONS 63
Instal·lació XEN 64
Procés d'Instal·lació XEN HYPERVISOR 4.0 64
Paquets a instal·lar 64
Posada en Marxa 64
Clonar màquina virtual 67
Màquina Virtual de Càlcul 67
Màquina Virtual de docència 69
Instal·lació Grid Engine 69

Aprofitament de Recursos Computacionals Estàndard per a la Creació d'un Clúster de Càlcul

En els temps de crisis econòmica i austeritat financera actuals, on les grans inversions en equipaments singulars per a la recerca i el desenvolupament estan fora de joc, cal explorar noves vies tecnològiques de baix cost per continuar aprofitant el talent de la comunitat científica catalana i espanyola.

Aquest projecte, amb pretensions molt més contingudes, explorarà la forma en que, a petita escala, es poden aprofitar recursos informàtics ja existents més enllà de l'ús regular d'aquests.

Concretament, assentarem les bases sobre com aprofitar la potència dels recursos computacionals d'un aula o aules de la Facultat de Matemàtiques per tal de crear un grid de càlcul d'ús pel personal docent i investigador de la facultat de matemàtiques.

Encara que aquest grid no podrà competir amb grans infraestructures de càlcul computacional existents en alguns departaments o facultats de la universitat, així com amb equipaments singulars de la ciutat, com el Barcelona Supercomputing Center o el Centre de Serveis Cientificotècnics de Catalunya, o estatals com el Centre de Supercomputació de Galícia, sí que servirà per a realitzar gran part dels càlculs previs o primeres simulacions bàsiques que s'utilitzen en l'àmbit de la recerca.

Per altra banda, aquesta plataforma també servirà per a formar nous investigadors en l'ús d'aquestes tecnologies, a més de permetre a alumnes que podrien ser futurs administradors, tècnics i responsables de sistemes treballar, practicar i entendre el funcionament dels mecanismes bàsics que componen aquestes estructures singulars.

Introducció als Sistemes de Càlcul d'Altes Prestacions

Els sistemes d'alt rendiment o d'altes prestacions, de l'anglès *High Performance Computing (HPC),* fan referència a una branca de la computació aplicada que es centra fonamentalment en la solució de problemes que fan un ús intensiu del càlcul.

Orígens

Fa uns pocs anys, els sistemes d'altes prestacions, també coneguts com sistemes de supercomputació, estaven dominats per sistemes grans, complexes i especialitzats que es trobaven principalment en els centres d'investigació més punters. No obstant, a mesura que la capacitat de càlcul dels sistemes estàndard ha augmentat, la relació entre el cost i el rendiment ha variat passant, doncs, les càrregues de treball a entorns de nivell equivalent als sistemes més usuals d'infraestructura de servidors.

Tipus de Càlcul d'altes prestacions

Computació paral·lela

La computació paral·lela és una tècnica de programació en la que moltes instruccions s'executen simultàniament, operant en base al principi de que sovint es poden dividir problemes grans en altres de més petits, els quals llavors es poden solucionar concurrentment ("en paral·lel"). Hi ha unes quantes formes diferents de computació paral·lela: a nivell de bit, a nivell d'instrucció, a nivell de dades, i a nivell de paral·lelisme de tasca. El paral·lelisme s'ha emprat durant molts anys, principalment en la computació d'alt rendiment, però l'interès en paral·lelisme ha augmentat últimament a causa de les restriccions físiques que eviten l'escalada de freqüència. Com el consum de potència (i conseqüentment la generació de calor) per ordinadors s'ha convertit en una preocupació durant els darrers anys, la computació paral·lela s'ha convertit en el paradigma dominant en l'arquitectura informàtica, principalment en forma de processadors multinucli.

Els ordinadors amb capacitat de computació paral·lela es poden classificar segons el nivell en el qual el maquinari dóna suport al paral·lelisme -amb ordinadors multinucli i ordinadors multiprocessador- que tenen elements de processament múltiples dins d'una única màquina, mentre que els clústers i els *grid* utilitzen ordinadors múltiples per fer feina en la mateixa tasca. Les arquitectures informàtiques paral·leles especialitzades s'utilitzen a vegades al costat de processadors tradicionals per a accelerar tasques específiques.

Els programes informàtics paral·lels són més difícils d'escriure que els seqüencials perquè la concurrència introdueix noves classes d'errors de programari potencials, dels quals les condicions de carrera són les més comunes. La comunicació i sincronització entre les subtasques diferents són alguns dels obstacles més grans per aconseguir el bon rendiment d'un programa paral·lel. El guany de velocitat d'un programa com a resultat de la paral·lelització és governat per la llei d'Amdahl[1].

Computació Distribuïda

La **computació distribuïda** és un nou model informàtic que permet fer grans càlculs utilitzant milers d'ordinadors i estalviant així els costos d'un superordinador. Aquest sistema es basa en repartir la informació a través d'Internet mitjançant un programari, prèviament descarregat per l'usuari, a diferents ordinadors que van resolent els càlculs i enviant els resultats al servidor.

Aquests projectes, quasi sempre solidaris, reparteixen la informació a processar entre milers d'ordinadors d'usuaris voluntaris per poder assolir quotes de processament a vegades més grans que les obtingudes per superordinadors.

El procés comença amb la descàrrega d'un programari informàtic d'un projecte de computació distribuïda. Aquest programari, normalment instal·lat com a servei dins del sistema operatiu, utilitza els recursos computacionals quan aquest no són usats per l'usuari.

El programari es connecta amb el servidor i obté el conjunt de dades o càlculs que haurà de resoldre i que un cop resolts, enviarà de nou al servidor per afegir aquella informació a la que han

[1] La llei d'Amdahl és el model que ens mostra la relació entre la velocitat de càlcul i la seva paral·lelització.

aportat altres usuaris. El servidor s'encarregarà de tractar conjuntament la informació recopilada als ordinadors dels usuaris.

Exemples d'aquest model de computació són el projecte SETI@home i el Folding@home.

Sistemes de Càlcul d'altes prestacions a l'estat Espanyol

Centre de Serveis Científics i Acadèmics de Catalunya (CESCA)

L'any 1991 la Generalitat de Catalunya va impulsar la creació d'aquest consorci a través de la Fundació Catalana per a la Recerca i la Innovació, amb la col·laboració de les universitats i del Consell Superior d'Investigacions Científiques (CSIC).

En els seus inicis, el CESCA tenia per objectiu proporcionar una infraestructura per actuar com a centre de supercomputació. Amb el pas del temps, aquest objectiu fundacional ha anat evolucionant i el CESCA ha expandit la seva activitat per aportar nous valors i serveis a la comunitat acadèmica i científica i, en general, al sistema d'R+D+I.

Actualment disposa de varis equips de computació que acumulen gairebé una capacitat de 4 TFLOPS (*Floating point Operations per Second)* des dels SGI Altix del tipus NUMA *(Non-Uniform Memory Acces)*. Fins als més recents Bull NovaScale amb més de 2,5 TFLOPS, amb la nova generació de processadors Xeon basats en l'arquitectura Nehalem d'intel.

Centro de Supercomputación de Galicia (CESGA)

El CESGA és el centre de càlcul, comunicacions d'altes prestacions i serveis científico-técnics de la comunitat científica gallega, impulsat pel sistema acadèmic universitari i pel CSIC.

Des de la seva posada en funcionament, el centre ha anat incorporant diferents supercomputadors, adaptant-se a les màximes prestacions en cada moment.

L'any 2007 va suposar un salt qualitatiu pel centre, ja que es va instal·lar el Finisterrae amb una capacitat de càlcul de 20 TFLOPS, col·locant-lo, en aquell moment, entre les 100 màquines més potents del món[2].

Barcelona Supercomputing Center (BSC)

El Centre Nacional de Supercomputació, també conegut com Barcelona Supercomputing Center, és un centre estatal localitzat a Barcelona. Està gestionat per un consorci format pel Ministeri d'Educació i Ciència, la Generalitat de Catalunya i la Universitat Politècnica de Catalunya. Forma part de la *Red Española de Supercomputación*, creada pel Ministeri de d'Educació i Ciència.

El centre va ser construït l'any 2005 a l'antiga capella anomenada Torre Girona. Dins les seves instal·lacions es troba el supercomputador Marenostrum[2].

[2] Es troba entre les 500 màquines de càlcul més potents segons el rànquing Top500.org

Elements Clau en Sistema HPC

Elements de Maquinari

En aquest apartat descriurem tots aquells elements físics necessaris per a la implementació d'un sistema de càlcul d'altes prestacions fent èmfasi en les diferències d'aquests mateixos elements en entorns més estàndards com poden ser l'entorn empresarial o el domèstic.

Elements de Càlcul

Definirem com Element de Càlcul, d'ara en endavant EC, com la mínima expressió de maquinari capaç per si mateixa d'elaborar un càlcul en sèrie. També analitzarem les possibles millores en cada un dels apartats de la configuració bàsica.

CPU

Una **Unitat Central de Procés** (UCP/CPU) anomenada col·loquialment com a ***processador*** és un component electrònic digital capaç d'interpretar instruccions de forma ordenada, de processar dades i generar la informació requerida. A la CPU s'executen les instruccions dels programes i es controla el funcionament dels diferents components de l'ordinador. Sol estar integrada en un xip anomenat microprocessador.

Està constituïda per dos unitats funcionals: la unitat aritmetico-lògica, i la unitat de control.

Els microprocessadors moderns estan integrats per milions de transistors i altres components empaquetats en una càpsula de grandària variable segons les necessitats de l'aplicació de la CPU i que van actualment des de la grandària d'un gra de llentia fins al de quasi una galeta. Les parts lògiques que componen un microprocessador són, entre altres:

- ***Unitat aritmetico-lògica*** (UAL o *A*rithmetic *L*ogical *U*nit) Realitza una operació segons l'opcode (operation code) indicat per la Unitat de Control. Aquesta operació pot ser aritmètica (Suma, Resta, Divisió, Multiplicació), Lògica (AND, OR, XOR...), o un desplaçament dels bits de la variable (shift).
- ***Unitat de control*** Unitat inclosa a la CPU encarregada de llegir les instruccions màquina guardades en la memòria principal i de generar les senyals de control necessàries per

controlar i coordinar la resta de les unitats funcionals d'un ordinador amb el propòsit d'executar les instruccions llegides. Són petites unitats que emmagatzemen dades del processador, la capacitat dels registres serà una o una altra. Per exemple en arquitectures de 64 bits, els registres són de 64 bits. En processadors Pentium, són de 32 bits. I en microcontroladors, acostumen a ser de 8 bits. Es poden implementar amb flip-flops o, actualment, amb fitxers de registres (*Static RAM*).

- ***Memòria cau*** Com que l'accés a memòria principal és molt lent en comparació amb les velocitats del processador, en processadors amb certes prestacions s'acostuma a usar una memòria a mig camí que manté una còpia de les dades de *memòria*, anomenada memòria cau, per suavitzar aquesta diferència de velocitats.

Els tipus d'arquitectures CPU més comuns:

- *RISC:(Reduced Instruction Set Computer)*

La filosofia dels processadors RISC va sorgir de la necessitat d'augmentar l'eficàcia de les CPU, està basada en la de l'arquitectura CISC i consisteix en treballar amb instruccions més senzilles, això permet que els processadors siguin més ràpids i eficients però també fa que les CPU depenguin de compiladors més complexos. Alguns models que l'usen són PowerPC (MAC) o SPARC.

- *CISC:(Complex Instruction Set Computer)*

Va ser la primera arquitectura de CPUs, els seus inicis es remunten a la dècada dels '60 i '70, és bastant econòmica, té pocs errors, la seva arquitectura és força complexa, el seu sistema de treball es basa en microprogramació i les instruccions son descodificades internament i executades amb una sèrie de microinstruccions emmagatzemades a la ROM. L'usen Intel i AMD a nivell extern.

Un processador single-threaded pot executar només una seqüència d'instruccions (*thread*) al mateix temps de forma successiva. Així aquest tipus de processadors són poc apropiats per a entorns multitasca tot i que degut a la dificultat per a serialitzar el disseny dels processadors HyperThreading han estat usats gairebé fins l'actualitat.

Un processador amb tecnologia HyperThreading pot executar en un sol bus de direccions almenys dues seqüències de forma paral·lela aconseguint aprofitar millor els recursos.

En l'actualitat els processadors amb tecnologia de multinucli (Quad Core en endavant) contenen 2 nuclis d'execució. Per tant s'aconsegueix l'execució paral·lela *real* de les dues seqüències. En executar 2 seqüències, cadascuna farà servir un nucli d'execució independent, concepte diferent al HyperThreading. Així, per exemple, un processador amb nucli doble i tecnologia HyperThreading de 2 branques permet executar 4 seqüències gairebé en paral·lel obtenint així un major rendiment.

GP/GPU

La Unitat de Procés Gràfic és un dispositiu dedicat a la generació de gràfics per a ordinadors personals, estacions de treball o consoles de videojocs. Les GPU modernes són molt eficients a l'hora de mostrar gràfics d'alta resolució gràcies a la seva forma paral·lela de treballar. La GPU usualment està integrada dins d'una *targeta gràfica* tot i que a vegades també pot estar integrada dins de la *placa base*.

Així, una GPU és capaç d'aplicar una sèrie d'operacions base sobre un gràfic de forma molt més ràpida que la CPU. Les operacions més comunes són dibuixar triangles, rectangles, cercles, arcs i l'operació BitBLT (combinar diversos bitmaps). Les GPU modernes inclouen funcions relacionades amb el vídeo digital i són capaces de suportar entorns 3D.

Les GPU modernes utilitzen la majoria dels seus transistors per a realitzar càlculs relacionats amb els gràfics d'ordinador en 3D. Inicialment van ser utilitzats per accelerar la memòria de treball intensiu de mapejat de textures i polígons de subproductes, per després afegir unitats per tal d'accelerar els càlculs geomètrics, com ara la rotació i translació dels vèrtexs en els diferents sistemes de coordenades. L'evolució recent de les GPUs suporten 'shaders' programables que poden manipular els vèrtexs i les textures amb moltes de les mateixes operacions amb el suport de la CPU, el sobremostreig i tècniques d'interpolació per reduir l'aliasing.

Tot i que ho podria semblar no és possible substituir la GPU per una CPU ja que una CPU, tot i tenir una major freqüència de rellotge, treballa d'una forma molt diferent. Les GPU s'estan

desenvolupant ràpidament gràcies a la seva especialització en els gràfics (només estan pensades per a una tasca) i estant optimitzades pel càlcul de valors en coma flotant usuals en entorns 3D a diferència de les CPU molt ineficients en aquesta tasca.

Moltes aplicacions gràfiques necessiten un alt nivell de paral·lelisme en tenir unitats fonamentals de càlcul (vèrtex i píxels) completament independents. Per tant, és una bona estratègia usar la força bruta de les GPU per a completar altres càlculs al mateix temps. Els models actuals de GPU solen tenir 6 processadors de vèrtex (que executen vèrtex shaders) i entre dos i tres cops més de processadors de píxels (que executen fragment shaders). Així una freqüència de rellotge d'uns 600 Mhz (estàndard actualment a les GPU i baixa en comparació a una CPU de per exemple 4 Ghz) es tradueix en una potència de càlcul molt major per a la GPU gràcies a l'arquitectura en paral·lel.

Així, la major diferència entre GPU i CPU és l'arquitectura. Les CPU solen usar un model de Von Neumann i en canvi les GPU es basen en el Model Circulant, un model que facilita el processament en paral·lel i la segmentació de les tasques.

És en aquesta línia que s'estan movent fabricants com NVIDIA, ja que representen un millor eficiència en càlcul sobre cert tipus de càlcul i aplicatius.

Tot i això en la majoria sistemes de càlcul d'alt rendiment, l'ús primordial de les GPU és el mostreig per pantalla dels aplicatius corrent sobre el Sistema Operatiu. O com a complement dels nodes de càlcul amb targetes PCI-Express dedicades a l'ús càlculs i aplicatius específics.

Memòria RAM

La memòria RAM, de l'anglès *Random Access Memory* (Memòria d'Accés Aleatori), és un tipus de memòria informàtica, caracteritzat per un accés directe en qualsevol ordre (aleatori) en un temps constant, sense distinció de la posició on es trobi la informació ni de la posició de l'anterior lectura. Avui en dia es produeixen mitjançant circuits integrats. La frase memòria RAM es fa servir sovint per a referir-se als mòduls de memòria.

Això contrasta amb altres mecanismes d'emmagatzematge, com les cintes o els discs magnètics i òptics, en què es depèn de la posició del capçal mòbil de lectura. En aquests dispositius, el

moviment triga més que la mateixa transferència de dades, i el temps d'accés depèn de la posició física del següent element.

La paraula RAM s'associa amb tipus de memòria volàtils (com els mòduls de memòria DRAM), on la informació es perd quan l'alimentació s'apaga. Notis que altres tipus de memòria també són RAM (d'accés aleatori), com la memòria ROM i un tipus de memòria flash anomenat NOR-Flash.

La particularitat en els sistemes de càlcul d'altes prestacions és la relació pactada entre els nuclis i la quantitat de memòria. Buscant relacions de 2Gb, 4Gb, etc... per nucli. Això és molt important en la compilació dels aplicatius que correran sobre aquesta plataforma.

L'altre particularitat és l'ús de memòries **ECC** com en l'àmbit empresarial, tot i una petita pèrdua en el rendiment. Usant logaritmes a prova de d'errors que poden detectar i restablir errors d'entre 1 i 4 bits.

NIC i NPU

Una targeta de xarxa, físicament, és una targeta d'expansió inserida dins l'ordinador amb una o més obertures externes per on és connectat el cable de xarxa.

A nivell conceptual, la targeta de xarxa, també anomenada adaptador de xarxa o NIC (*Network Interface Card*), permet la comunicació entre els diferents dispositius connectats entre sí i també permet compartir recursos entre dos o més equips (discs durs, CD-ROM, impressora, etc). Hi ha diversos tipus d'adaptadors en funció del tipus de cablejat o arquitectura que s'utilitzi a la xarxa (coaxial fi, coaxial gruixut, Token Ring, etc.), però actualment el més comú és l'Ethernet, que utilitza una interfície o connector RJ-45.

Cada targeta de xarxa té un número d'identificació únic de 48 bits en hexadecimal anomenat direcció MAC. Aquestes direccions de maquinari úniques són administrades per l'Institute of Electronic and Electrical Engineers (IEEE). Els tres primers octets del número MAC són coneguts com a OUI i identifiquen a proveïdors específics i són designats per la IEEE.

S'anomena també NIC al xip de la targeta de xarxa que s'encarrega de servir d'interfície d'Ethernet entre un medi físic. És un xip utilitzat a ordinadors o perifèrics com les targetes de xarxa, impressores en xarxa o sistemes amb permís per connectar dos o més dispositius entre si a través d'algun medi, ja sigui una connexió sense fils, un cable UTP, un cable coaxial, o un cable de fibra òptica.

En els sistemes d'alt rendiment trobem definicions i topologies de xarxa específiques com poden ser 10 Gigabit, Myrinet o Infiband aquestes seran analitzades en més profunditat en l'apartat de xarxes.

Scratch

En aquells sistemes on degut a la topologia de xarxa és impossible treballar amb nodes de càlcul sense disc (HD less). S'usa el disc local per allotjar el sistema operatiu bàsic del node, així com les configuracions bàsiques.

Això representa un ús molt baix d'aquest disc, quan un càlcul té necessitat d'escriure temporalment al disc dur, s'usa el del mateix node per tal de no col·lapsar la xarxa amb transaccions de dades parcials.

En certes topologies de xarxa on la velocitat i la fiabilitat, així com un sistema de fitxers distribuïts d'alt rendiment, ho permeten. Hi ha la possibilitat d'eliminar dels nodes el disc dur. Capturant el sistema operatiu, les aplicacions en xarxa, i el disc d'Scratch com si fossin ubicades en local.

Elements de Xarxa

Aquí definirem les principals topologies de xarxa usades en els sistemes de càlcul d'altes prestacions. Les topologies més habituals de xarxa que trobem són:

- **Ethernet:** és una família de tecnologies basades en el marc de les xarxes d'ordinadors per a xarxes d'àrea local (LAN). El nom prové del concepte físic de l'èter. Es defineix una sèrie d'estàndards de cablejat i senyalització de la capa física del model OSI de xarxa, a través dels mitjans d'accés de xarxa al Media Acces Control (MAC)/ Capa d'enllaç de dades i un format d'adreces comú. Ethernet va ser estandarditzada com IEEE 802.3 que determina

com les màquines de la xarxa envien i reben dades sobre un medi físic compartit que es comporta com un bus lògic, independentment de la seva configuració física. Ethernet va ser desenvolupada als anys 70 al Xerox PARC, per Robert Metcalfe. Actualment Ethernet és l'estàndard més utilitzat en xarxes locals. Des dels anys 1990, s'utilitza freqüentment Ethernet per a les connexions entre clients. Aquesta configuració ha desplaçat altres estàndards com Token Ring, FDDI i ARCANET. En supercomputació en xarxes, tant de dades com de càlcul, cal com a mínim l'ús de tecnologia Gigabit Ethernet i actualment la tendència avança sobre els 10 Gigabit amb rendiments superiors als antics enllaços de fibra òptica. Fins i tot està previst en els propers dos anys l'aparició de les primeres xarxes a 40 Gigabit amb rendiments equivalents i fins i tot superiors a xarxes infiniband.

- **Myrinet:** és una xarxa d'interconnexió d'altes prestacions, desenvolupada per Myricom des de 1995. L'estructura física de Myrinet consisteix en dos cables de fibra òptica, upstream i downstream, encapsulats en un únic connector. La interconnexió es realitza mitjança encaminadors i switchos. Les últimes versions desenvolupades, Myri-10G, arriben a assolir amples de banda de 10Gbps i latències de 3 microsegons, arribant a ser compatibles i operables amb el protocol 10 Gigabit Ethernet. Altres dels seus avantatges són l'ús de processadors integrats específics per a les comunicacions, descarregant de feina els processadors del node de càlcul.
- **Infiniband:** és un bus de comunicacions sèrie d'alta velocitat, la versió més extrema és capaç d'assolir latències de milisegons i amples de banda bruts de fins a 120 Gbps. És per això que s'ha estès àmpliament en els entorns d'altes prestacions. Infiniband fa ús d'una topologia commutada de forma que varis dispositius poden usar la xarxa al mateix temps. Les dades són transmeses en paquets de 4kb agrupant-se en missatges. Un missatge pot ser una operació d'accés directe a memòria de lectura o escriptura sobre un node remot, un enviament o recepció pel canal, una operació de transacció reversible o un multicast.

Xarxa de Càlcul

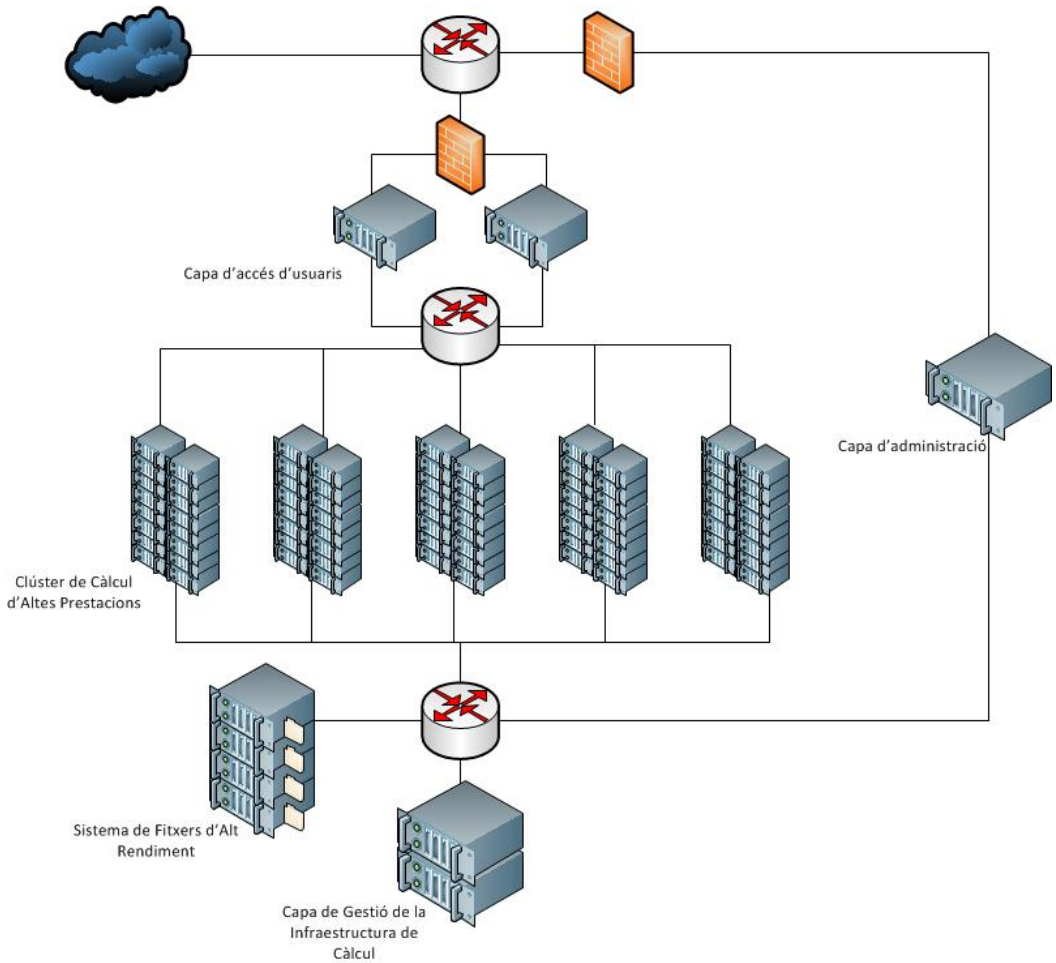
Aquesta xarxa està dedicada exclusivament a l'enviament de dades de càlcul. Optimitzant així els codis de paral·lelització com MPI. Usant directament les dades en memòria RAM o scratch a compartir entre els nodes.

Xarxa de Dades

Aquesta xarxa està optimitzada per a la comunicació amb el sistema de fitxers en xarxa. Aquí es fan totes les transaccions finals o parcials, així com la càrrega de les variables d'entorns, compiladors i aplicatius.

Xarxa d'Accés

Aquesta xarxa s'utilitzarà per a l'administració i debuggin en els nodes i els diferents elements del sistema.



Un dispositiu d'emmagatzematge pot suportar informació, informació de processament o ambdues. Un dispositiu que tan sols conté informació és un medi de gravació. Els dispositius on la informació de procés pot tant accedir a un medi de gravació separable i portable (esborrable), com a un component permanent per a emmagatzemar i recuperar informació es diuen **equips d'emmagatzematge de dades**.

- **Xarxa d'àrea d'emmagatzematge** en anglès **SAN** (Storage Area Network) és una arquitectura de xarxa per a connectar *dispositius d'emmagatzenatge informàtic* (com matrius de discs arrays), biblioteques de cintes i dispositius òptics a servidors, de manera que pel sistema operatiu els dispositius apareixen com connectats localment.

  Les SAN estan basades en tecnologia Fibre Channel i més recentment en iSCSI, a més dels sistemes de fitxers distribuïts que suporten protocols de xarxa com Myrinet o Infiniband. La seva funció és la de connectar de manera ràpida, segura i fiable els distints elements que la constitueixen. Encara que el seu cost i complexitat està disminuint, encara ara, les SAN encara són estranyes fora de les grans empreses.

  En contrast a les SAN, les NAS (Network Attached Storage) utilitzen protocols basats en arxius com NFS o SMB/CIFS on és evident que l'emmagatzematge és remot, i les computadores sol·liciten una porció d'un arxiu abstracte més que un bloc de disc.

  Dins del món de la computació d'alt rendiment, l'ús de sistemes de fitxes abraça des dels protocols SMB, fins a sistemes de fitxers distribuïts amb protocols propis i optimitzats per aquest ús. Com poden ser Lustre o Globla File System, a més del desenvolupament de protocols com el paral·lel NFS.

Sistemes de fitxers distribuïts d'alt rendiment

En els clústers d'alt rendiment, l'emmagatzematge massiu es gestiona a través de sistemes de fitxers especialitzats i desenvolupats específicament per a aquest entorn, com són:

- **LUSTRE:** és un sistema de fitxers distribuït de codi lliure, usat en clústers a gran escala. El nom prové de la barreja de Linux i clúster. El projecte intenta proporcionar un sistema d'arxius per a clústers amb capacitat de gestió de varis milers de nodes amb rangs de petabytes de capacitat d'emmagatzematge, sense perjudicar la velocitat d'accés a les dades i la seguretat, estant disponible sota GNU GPL.

  Entre d'altres companyies que han desenvolupat lustre cap a entorns propis, com poden ser HP, DELL, SUN, SGI... Els dissenyadors originals, desenvolupadors i mantenidors de la versió Open Source de lustre són Cluster File Systems.

  El seu funcionament es basa en la idea de que cada fitxer és un objecte per Lustre. Presentant a tots els clients una semàntica POSIX estàndard així com accés concurrent lectura i escriptura per a objectes compartits. El sistema es divideix en:

    - Meta data server (MDS): És desant les metadades dels fitxers.
    - Object storage target (OST): Element real d'emmagatzematge dels objectes.
    - Object storage server (OSS): Permet realitzar les gestions sobre els objectes emmagatzemats a l'OST.
    - Finalment el client, distribuït en tots aquells sistemes que hi hagin d'accedir.

  Tots aquest elements aconsegueixen que Lustre sigui capaç de moure dades directament entre l'aplicatiu i l'OSS de Lustre sense necessitat de realitzar un copia de les dades a través del nucli, aconseguint una baixa latència i un gran ample de banda en accés directe dels processadors al sistema de fitxers.

- **Global File System (GFS):** GFS i la seva evolució GSF2 s'assemblen en Lustre i difereixen dels sistemes de fitxers convencionals en l'accés directe i concurrent de tots els nodes al mateix bloc d'emmagatzematge.

  GFS però, no disposa de clients ni de servidors, ja que tots els nodes són *peers* o pares. L'ús de GFS requereix maquinari tipus SAN que permeti l'accés a l'emmagatzematge compartit, i un controlador que controli els accessos a aquests. El controlador opera com un element independent, tot i que GFS i GFS2 usen configuracions DLM (*Distributed Lock Manager*) per a sistemes en clúster.

Tant GFS com GFS2 són software lliure, distribuïts sota el terme de GNU General Public License.

Emmagatzematge Temporal

Està basat en l'ús del scratch local o distribuït, per tal de desar els resultats parcials i/o checkpoints dels diferents càlculs llençats al clúster.

Emmagatzematge Final

Recolzat en una SAN amb sistemes de fitxers com poden ser Lustre, GFS o senzillament un poc recomanable sistema Samba. Hi ha tots el fitxers de càlcul i resultats, ordenats per usuaris amb els pertinents permisos i mesures de confidencialitat.

Elements de Programari

En aquest apartat descriurem tots aquells elements de programari necessaris per a la implementació d'un sistema de càlcul d'altes prestacions.

Sistemes Operatius

El sistema operatiu és el programari responsable de gestionar els recursos en un terminal (ja sigui un ordinador personal, un telèfon mòbil, etc.). El sistema operatiu actua com a amfitrió dels diversos programes d'aplicació que normalment corren sobre una màquina. Una de les principals funcions és gestionar els detalls de l'operació del maquinari, de manera que els diversos programes no se n'hagin d'ocupar, alleugerint i fent més fàcil així el procés de programació d'aquestes aplicacions.

Els sistemes operatius més habituals en l'entorn del càlcul d'altes prestacions són basats en UNIX o Linux, tot i l'intent de Microsoft de desenvolupar una versió de SO basat en Windows Server 2008 especialitzada en HPC:

- Debian és una distribució conformada per una comunitat de desenvolupadors i usuaris que mantenen aquest sistema operatiu GNU. Aquesta comunitat és molt activa en tots els àmbits de la informàtica, tot i que no té suport oficial per part de cap empresa.

- SUSE Linux va començar sent una distribució comercial d'una empresa Alemanya amb el mateix nom. Aquesta empresa va ser fundada l'any 1992. A l'any 2001, però, SUSE Linux va entrar en crisi i va ser comprada per Novell. Aquesta empresa ha canviat el model de negoci de SUSE Linux i ha aconseguit fer-la altre cop rendible.
- Red Hat és famós a tot el món pels seus esforços destinats a promoure el programari lliure. No només treballen en el desenvolupament de les distribucions més populars de GNU/Linux, sinó també en la comercialització de diferents productes i serveis basats en programari lliure. Així mateix, tenen una gran infraestructura amb més de 500 empleats a 15 indrets del món.

  Desenvolupadors de Red Hat han desenvolupat múltiples paquets de programari lliure, els quals han beneficiat tota la comunitat. Algunes d'aquestes contribucions han estat la creació d'un sistema d'empaquetament de programari (RPM) i vàries utilitats d'administració i configuració d'equips, com *sndconfig* o *mouseconfig*.

- Microsoft Windows són una sèrie de sistemes operatius i interfícies gràfiques d'usuari produïts per Microsoft. Microsoft va introduir per primera vegada un entorn operatiu anomenat *Windows* el novembre de 1985 com un complement a MS-DOS, en resposta al creixent interès en les interfícies gràfiques d'usuari (GUI). Microsoft Windows ha arribat a dominar el mercat del PC, superant al Mac OS, que havia estat introduït prèviament. A partir d'octubre de 2009, Windows tenia aproximadament el 91% de la quota de mercat dels sistemes operatius client usats a Internet. La versió més recent d'un Windows client és el Windows 7; la versió més recent de servidor és el Windows Server 2008 R2, de la qual deriva Windows Server HPC 2008 per a l'ús de clúster d'alt rendiment.

Gestors de cues

El grid computing és una tecnologia innovadora que permet utilitzar de forma coordinada tot tipus de recursos (entre ells còmput, emmagatzematge i aplicacions específiques) que no estan subjectes a un control centralitzat. En aquest sentit és una nova forma de computació distribuïda, en la qual els recursos poden ser heterogenis (diferents arquitectures, supercomputadors, clústers...) i es troben connectats mitjançant xarxes d'àrea extensa (per

exemple Internet). Desenvolupat en àmbits científics a principis de l'any 1990, la seva entrada al mercat comercial seguint la idea de l'anomenada Utility computing suposa una gran revolució.

Per controlar tot aquest nombre de maquinari ja sigui distribuït en una LAN o en una WAN, cal un sistema de cues. Els més usuals són:

- **Sun Grid Engine (SGE):** és un sistema de cues de processos distribuïts de codi obert desenvolupat per Sun MicroSystems. Com el seu nom indica l'objectiu final d'aquest software és la posada en marxa i gestió d'una arquitectura de graella.

  SGE s'usa freqüentment en els sistemes de càlcul d'alt rendiment, encarregant-se de l'acceptació, programació, enviament i gestió de l'execució remota i distribuïda d'un gran nombre de treball en espais d'usuari individuals, paral·leles o interactives. Entre les funcionalitats de SGE també es disposa de la possibilitat de gestionar i programar la reserva i la prioritat dels recursos distribuïts com processadors, memòria, espai d'emmagatzematge i llicències de programari.

- **TORQUE/MAUI (PBS):** sistema basat en OPENPBS. És un sistema que gestiona recursos de computació, implementa l'ús de les cues per a l'execució de treballs en el clúster, a més gestiona els recursos distribuïts com processadors, memòria, espai... Per altra banda MAUI fa de planificador en el que es defineixen l'assignació d'aquests recursos maximitzant el rendiment de TORQUE.

Parel·lització

a computació paral·lela s'ha convertit en el paradigma dominant en l'arquitectura informàtica, principalment en forma de processadors multi nucli.

Una aplicació construïda amb el model híbrid de programació paral·lela pot executar-se en un clúster d'ordinadors fent servir OpenMP i Message Passing Interface (MPI).

- **OpenMP (Open Multi-Processing):** és una interfície de programació d'aplicacions (API) que suporta programació multiprocés amb memòria compartida multi-plataforma en C/C++ i Fortran a moltes arquitectures, incloent les plataformes Unix i Microsoft Windows. Consisteix en un conjunt de directives de compilador, rutines de biblioteques, i variables d'entorn que afecten al comportament en temps d'execució.

  Definit conjuntament per un grup dels principals fabricants de maquinari i programari, OpenMP és un model portable i escalable que dóna als programadors una interfície simple i flexible per a desenvolupar aplicacions paral·leles per a plataformes que van des de l'escriptori fins als supercomputadors.

- **MPI (Message Passing Interface):** és un dels estàndards que defineix una sintaxis i una semàntica per les funcions contingudes en una biblioteca de transmissió de missatges per sistemes amb múltiples processadors i cores.

  La interfase MPI disposa d'implementacions en multitud de llenguatges com poden ser C, C++, Fortran i Ada. El principal avantatge és que els programaris que usen la biblioteca són portables i ràpids, ja que estan optimitzats pel hardware sobre el que corren.

Compiladors

Un compilador és un programa d'ordinador que tradueix un llenguatge informàtic, com el Visual Basic o el C, per exemple, a un altre llenguatge informàtic. La tasca típica d'un compilador és la traducció d'un llenguatge d'alt nivell a un altre (normalment Assemblador) de baix nivell. Cadascun dels processadors que existeixen tenen una versió pròpia d'un conjunt d'instruccions al qual tradueixen els compiladors. Aquesta eina permet al programador desconèixer el llenguatge que utilitza l'ordinador i escriure en un llenguatge més universal i més proper a com pensa un humà.

El Fortran va ser el primer llenguatge d'alt nivell que va comptar amb un compilador. Avui en dia hi ha un gran nombre de llenguatges informàtics que poden ser compilats

Els compiladors és poden dividir en:

- Compiladors de GNU: gcc, g77, fortan...
- Compiladors Intel: Icc, Ifort...
- Compiladors AMD: porlan (PGI), ACML,...
- Compilador SUN (sparc): Sun Studio, ...

Programari de gestió i control

Programari per a la monitorització i control de l'estat del sistema:

- **Nagios** és un sistema de monitorització de xarxes de codi obert molt estès. Controla equips i programari, alertant-ne quan el comportament no és el desitjat. Entre les seves principals característiques apareix el suport per al control de serveis de xarxa (SMTP, POP3, HTTP,...), monitorització de sistemes de maquinari (carga del processador, ús de disc, memòria, estat dels ports, etc...) amb independència de sistemes operatius, i amb la possibilitat de monitorització remota xifrada amb SSL i SSH.
- **Ganglia** és una distribució de software per a la monitorització del sistema i xarxes. Permet controlar rendiments i guardar estadístiques i històrics. S'instal·la com a Daemon de linux i consta de suport multiplataforma.

Consideracions Tècniques

A continuació farem un resum i explicació de les parts comunes en totes les solucions possibles, explicant la gran majoria de possibilitats de configuració així com definint i parametritzant els elements claus dels sistemes de les solucions.

Per altra banda cal mencionar també la possibilitat de realitzar tot un seguit de solucions intermèdies entre les solucions les quals seran comentades amb una visió més superficial.

Sobre els serveis actuals s'estudiarà garantir la integració amb tots aquells serveis necessaris per a que l'alumnat pugui seguir realitzant les seves labors sense interferir-hi. Serveis com l'autenticació de l'ús de màquines Linux i Windows i l'accés a les comptes d'usuari hauran de ser possibles després de la instal·lació de l'aula.

Infraestructura de Servidors

Maquinari

El sistema de maquinari que cal per suportar aquesta solució abraça des de la configuració més senzilla amb un únic servidor fins a solucions més avançades amb més d'un servidor per gestionar les càrregues del sistema a més dels accessos dels usuaris, així com una millor gestió de les dades mitjançant solucions d'emmagatzematge i còpies de seguretat sobre les dades del sistema.

Programari

El sistema operatiu escollit per la solució, a nivell de servidor, estarà sempre basat en l'última distribució estable de debian.

Dins d'aquest servidor o infraestructura de servidors hi haurà l'estructura de directoris d'usuari pròpia del clúster. D'aquesta forma els usuaris amb accés al càlcul hauran d'accedir via SSH a la màquina per carregar els paquets i compiladors necessaris mitjançant l'eina modules i enviar els càlculs al clúster mitjançant el gestor de cues.

Segons els recursos hardware disponibles els rols de cada un dels servidor seran explicats més endavant.

Cal tenir en compte que en totes les solucions no hi haurà integració de sortida amb LDAP de la universitat. La possibilitat de realitzar la integració es podria treballar més endavant com una millora. A més aquest recurs de càlcul estarà limitat al PDI de la facultat de matemàtiques.

Utilitzarem el CRON per tal d'iniciar les màquines en mode de càlcul i reiniciar-les en mode de treball.

Les màquines obtindran la IP de la xarxa de càlcul i administració mitjançant un DHCP en aquest servidor. Les màquines virtuals de docència obtindran la IP mitjançant DHCP propi de la UB mitjançant una xarxa de la UB. Veure més endavant.

El llistat de programari instal·lat en aquesta part de la infraestructura serà (no contemplem tots aquells paquets secundaris necessaris per al funcionament d'ells mateixos):

- COMPILADORS
    - GCC, g77, ifort, icc, porland...
- MODULES
- SUN GRIN ENGINE (Gestor de Cues + Scheduler)
- GANGLIA (Monitorització)
- Biblioteques de Parl·lelització
    - MPI, MPICH
- Biblioteques Matemàtiques
    - BLAS, LIBGOTO, ATLAS
- Aplicacions de Càlcul Optimitzades
- Distributed Shell (permet replica comandes ubicats al fitxer /etc/dsh/machines.list)

Cal recordar que durant l'horari de docència els treballs quedaran en cua per tal de poder executar-los a la nit.

Infraestructura d'Aula

Maquinari

El requeriment principal a nivell de maquinari dels ordinadors és que el processador sigui compatible amb la tecnologia VT-X[3] d'intel o AMD-V, això és necessari pel correcte funcionament de XEN.

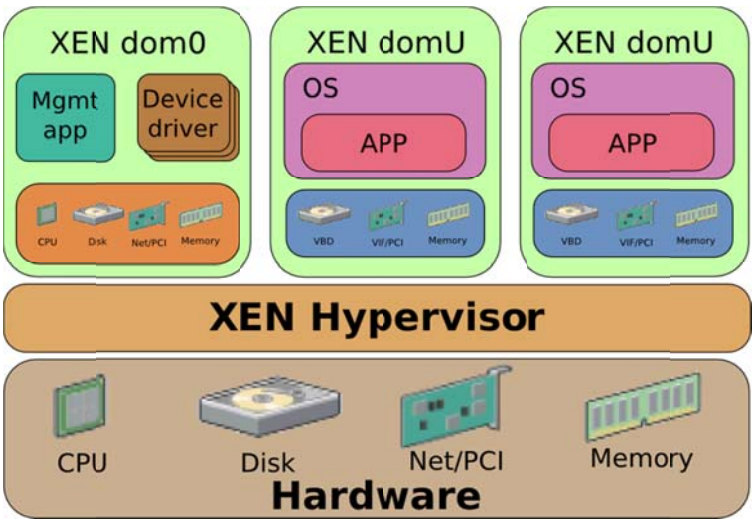
Altres requeriments importants són la disponibilitat d'una partició dedicada al Scratch així com un targeta de xarxa gigabit.

Programari

Paravirtualització amb XEN

Xen és una màquina virtual de codi obert desenvolupada per la Universitat de Cambridge. La meta del disseny és poder executar instàncies de sistemes operatius amb totes les seves característiques, de forma completament funcional en un equip senzill.

Xen proporciona un aïllament segur, control de recursos, garanties de qualitat de servei i migració de màquines virtuals en viu. Els sistemes operatius han de ser modificats explícitament per córrer Xen (encara que mantenint la compatibilitat amb aplicacions d'usuari). Això permet a Xen assolir virtualització d'alt rendiment sense un suport especial de maquinari.



[3] Ref: http://ark.intel.com/search/advanced?VTX=true

Hypervisor Nadiu

L'Hypervisor Nadiu en paravirtualització és el programari que s'executa directament sobre la capa de maquinari.

Aquest model representa la virtualització clàssica on l'Hypervisor gestiona tots els recursos de maquinari disponibles i controla els sistemes operatius hostejats.

DOM0

El dom0, o domini zero, és el primer domini arrancat per l'Hyperviosr en arrancar. Aquest domini té privilegis propis, com la possibilitat d'arrancar nous dominis, i és capaç d'accedir directament a tots els recursos de maquinari. És també el responsable de carregar tot els controladors dels dispositius de maquinari, a menys que es creïn dominis específics per aquesta funció.

Dom0 és l'encarregat, com a BackenDriver, de multiplexar i redireccionar totes les peticions de maquinari en els controladors FrontenDriver a cada domU.

DOMU

Un domU és el complementari del dom0; són aquells dominis sense privilegis d'accés al maquinari. Han d'accedir mitjançant a FrontendDriver per la multiplexació del maquinari, el qual comparteixen amb els altres dominis. Un domU és arrancat pel procés xen des del dom0, accessible per l'usuari. El Kernel dels domU s'hereta del sistema de fitxers del propi dom0, no des del sistema de fitxers dels domU.

Màquines Virtuals

Generarem 2 domU, un per cada tipus de màquina virtual necessària.

Màquina de Càlcul

Un màquina virtual Debian estable serà recurs de càlcul actiu al sistema de cues.

L'ús de les comandes `start` and `resume` del Xen permetrà teòricament congelar l'estat del càlculs fent possible reprendre'ls a la nit següent.

Aquestes comandes hauran de tenir replicada l'estructura d'usuaris dels servidors així com les claus SSH per tal de permetre l'ús de les interfícies MPI i poder paral·lelitzar els càlculs entre diverses màquines.

Disposaran dels clients de Ganglia per tal de poder monitoritzar des del servidor tots els recursos de maquinari.

Màquines de docència

Les màquines virtuals de docència són tractades com a caixes negres pel sistema, ja que en essència són una màquina virtual amb una estructura determinada, en aquest cas Windows i Linux.

La seva instal·lació no suposa, en principi, problemes amb sistemes com REMBO ja que en disposar directament d'una IP pròpia, les imatges podran ser distribuïdes de forma estàndard.

Més endavant descriurem com seran integrades aquestes màquines en la infraestructura i la possible integració amb les serveis de la universitat i els alumnes.

Els problemes més usuals que podem trobar són una pèrdua en el rendiment de la pròpia màquina virtual sobre la mateixa versió física. Així com problemes de renderització i acceleració gràfica ja que la GPU treballarà com un recurs compartit.

Una possible solució per a aquest aspecte seria permetre la càrrega i distribució per PXE d'un sistema operatiu en RAM amb la optimització pertinent per l'entorn gràfic o senzillament reservar i mantenir fora de la infraestructura les aules amb requeriments d'aquest tipus.

Scratch

Caldrà que els ordinadors disposin d'un disc o partició d'espai per l'scratch la qual haurà de ser visible per la màquina de càlcul i separada física i lògicament de la partició de XEN.

Podríem trobar problemes de rendiment en sobreescriure constantment dins de les màquines virtuals ja que al cap i a fi poden ser considerades com fitxers dins d'uns sistemes de fitxers.

La solució òptima passa per crear una partició o, en el millor dels casos, un disc dedicat al scratch.

La gestió de les màquines virtuals serà a través del servidor amb l'ús del crontab i scripts. Amb aquest sistema també serem capaços de gestionar l'energia dels equips, ja que podrem encendre i apagar les màquines mitjançant Wake-On-Lane.

A diferència dels servidors el ordinadors de sobretaula, la majoria de les estacions de treball no tenen accés a través de ILO amb protocol IPMI per poder accedir a la bios, consola o comandes d'arrancada o aturada. Per aquest motiu haurem de tenir inventariades totes les adreces MAC de les màquines de docència per tal de poder usar la comanda etherwake

```
Script WOL.exe

etherwake A4:55:25:H4:3G:ED
etherwake A4:55:25:65:3D:13
etherwake ...
```

El script starthpc.exe i resumehpc.exe seran els encarregats de fer el `start, suspend` i el `resume` de les màquines de càlcul així com l'arrencada i aturada de la màquina de docència.

```
Script starthpc.exe

for i in `cat hostnames.txt`; do
ssh $i xm shutdown docencia
ssh $i xm start /etc/xen/calcul/calcul.cfg
done

Script suspendhpc.exe
for i in `cat hostnames.txt`; do
ssh $i xm suspend calcul
ssh $i xm create /etc/xen/docencia.cfg
done

Script resumehpc.exe

for i in `cat hostnames.txt`; do
ssh $i xm shutdown docencia
ssh $i xm resume calcul
done
```

Tenim l'opció d'inserir aquests scripts al servidor o tots al dom0 sent preferible el primer, ja que permetrà una gestió d'aquest més senzilla a l'hora d'efectuar canvis.

Per altra banda, per defecte les màquines de la sala arrancaran amb la màquina de docència per raons de seguretat en cas de falla del sistema de càlcul, assegurant així el correcte ús de la sala en cas de problemes.

La configuració per defecte de Xen arrenca totes les màquines ubicades a `/etc/xen/*.cfg` després d'iniciar el dom0. Per tant, per evitar l'inici de la màquina de càlcul haurà d'estar ubicada a `/etc/xen/calcul/calcul.cfg`.

Gestor de Cues

Els gestors de cues o batch queue systems són aplicacions històricament desenvolupades per a l'execució de programes batch.

Aquests sistemes els podem trobar dins de la gran majoria de Sistemes Operatius amb la funció de gestionar, d'una forma bàsica el Job Scheduling dels processos, serveis i aplicacions que corren sobre el mateix sistema.

Dins l'entorn de la computació d'altes prestacions s'han desenvolupat cluster batch systems per planificar l'execució de programes i scripts, les diferents cues i els recursos de maquinari de que disposa el sistema.

Alguns dels sistemes més coneguts i extensament usats són Oracle Grid Engine, Open Grid Scheduler, Univa Grid Engine, Sons of Grid Engine, SLURM o Torque/Maui.

Grid Engine Project

Conegut en els seus inicis com CODINE (COmputing in DIstributed Networked Environments) és un gestor de cues adquirit per Sun Microsystems, reanomenat SUN Grid Engine, del qual va publicar el seu codi sota llicència SISSL durant el 2001 per al seu desenvolupament.

La compra de SUN per part d'Oracle el 2010, així com la renúncia per seguir desenvolupant el programari, ara anomenat Oracle Grid Engine, dins del marc open-source va provocar l'aparició per part de la comunitat open-source de diferents bifurcacions del projecte sorgides de l'última

versió open-source publicada per SUN la SGE6.2u5. Les més conegudes són Univa Grid Engine, Son of Grid Engine i Open Grid Scheduler.

Aquests sistemes funcionen mitjançant una sèrie de comandes que permeten enviar jobs detallant les necessitats com la memòria, cputime, quantitat de disc dur, nombre de nuclis, software, entorn paral·lels, etc...

Aquest treballs són registrats pel gestor, que determina la disponibilitat dels recursos disponible i tramita el job quan aquests estan disponibles segons unes regles d'allotjament (prioritat, urgency, quotes, etc...)

Els usuaris disposen de la informació relativa als seus jobs, així com permisos per eliminar-los.

Els principals avantatges de Grid Engine són:

- Integració de Batch Queue System i Scheduler en el mateix paquet
- Estadístiques sobre els Jobs
- Informació Detallada dels recursos de càlcul
- Permet la suspensió, reactivació i migració de jobs
- Integració amb elements de Checkpointing
- Integració d'entorns paral·lels
- Job arrays
- Polítiques de compartició de recursos molt detallades i acurades
- API's per a la integració i desenvolupament de softward 3rd party
- Suport per entorns GPU

Open Grid Scheduler

Rols dels Nodes

Host: Entenem per host o node tots el equips que formaran part del Clúster del Gestor de Cues i tindran algun dels rols definits per aquest. Cada tipologia de host desenvolupa una tasca concreta dins del clúster governat pel gestor. Un host en alguns casos pot desenvolupar diversos rols.

Master Host: El master host és el host encarregat de gestionar tota l'activitat del clúster. Aquest node corre un daemon anomenat `sge_qmaster`, el qual controla tots els components del grid com les cues i els jobs. A més, també manté actualitzades taules dels recursos i reporta tota la informació relativa als usuaris i la seva activitat. Aquest node també corre scheduler o planificador `sge_schedd`. Aquest node no requereix cap mena de configuració més enllà que la inicial durant el procés d'instal·lació del Open Grid Engine.

Shadow Master Host: Aquest són tots aquells nodes del clúster preparats per, en cas de detectar una fallida al master host, prendre el rol d'aquest i la governança del sistema.

Quan un shadow master host detecta que el master daemon `sge_qmaster` es comporta de forma anormal, arranca una nou `sge_qmaster` en aquest node.

Aquest node corre el daemon `sge_shadowd`, el qual conté tota la informació relativa al `sge_qmaster`. El shadow master ha de compartir, i per tant tenir accés, al `sge_qmaster` per tal de tenir visibilitat de l'estat dels jobs, recursos, configuració de cues i usuaris. Per tant, ha de disposar d'accés, tant de lectura com d'escriptura, a tot el conjunt de directoris del sge al master host i accés `al sge-root/cell/common`.

Així com garantir l'accés compartit mitjançant NFS o sistemes de redundància de fitxers `al sge_qmaster`.

Execution hosts: Aquest són els nodes que tenen permisos per executar jobs. Els execution host són per tant els nodes que allotgen les instàncies de les cues i corren el daemon `sge_execd`.

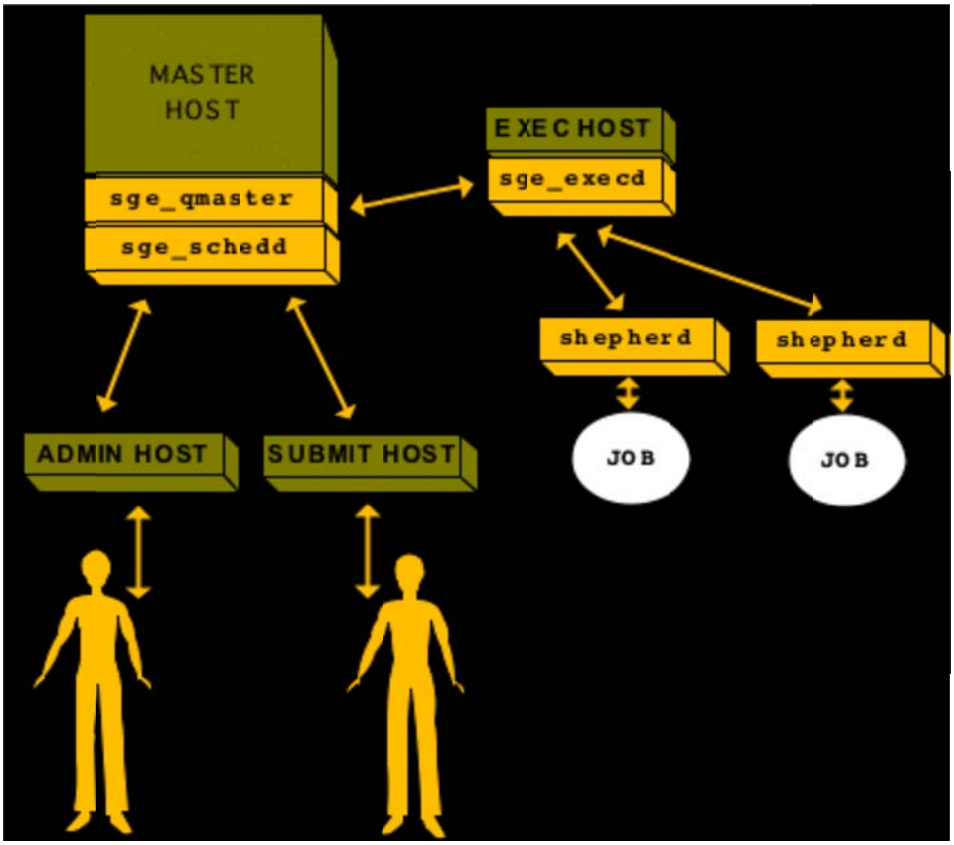
Administration Hosts: Podem permetre l'administració i configuració del clúster a diferents nodes anomenats Administration Host. És important entendre que no allibera de cap càrrega als Master Host si no que tan sols ens permet realitzar l'administració del clúster (relativa a afegir nodes, cues, usuaris i jobs) a diferents nodes.

Els administrative host són afegits al sistema mitjançant la comanda:

`qconf -ah` *`hostname`*

Submit Hosts: Els submit Hosts són tots aquells nodes que tindran permisos per enviar i controlar batch jobs. Els usuaris mitjançant loggin en aquest host i l'ús de la comanda `qsub` podran enviar jobs al sistema. Per altra banda, des d'aquests nodes els usuaris podran controlar l'estat dels jobs mitjançant la comanda `qstat` i córrer d'interfície gràfica d'usuari `QMON`.

`qconf -ah` *`hostname`*



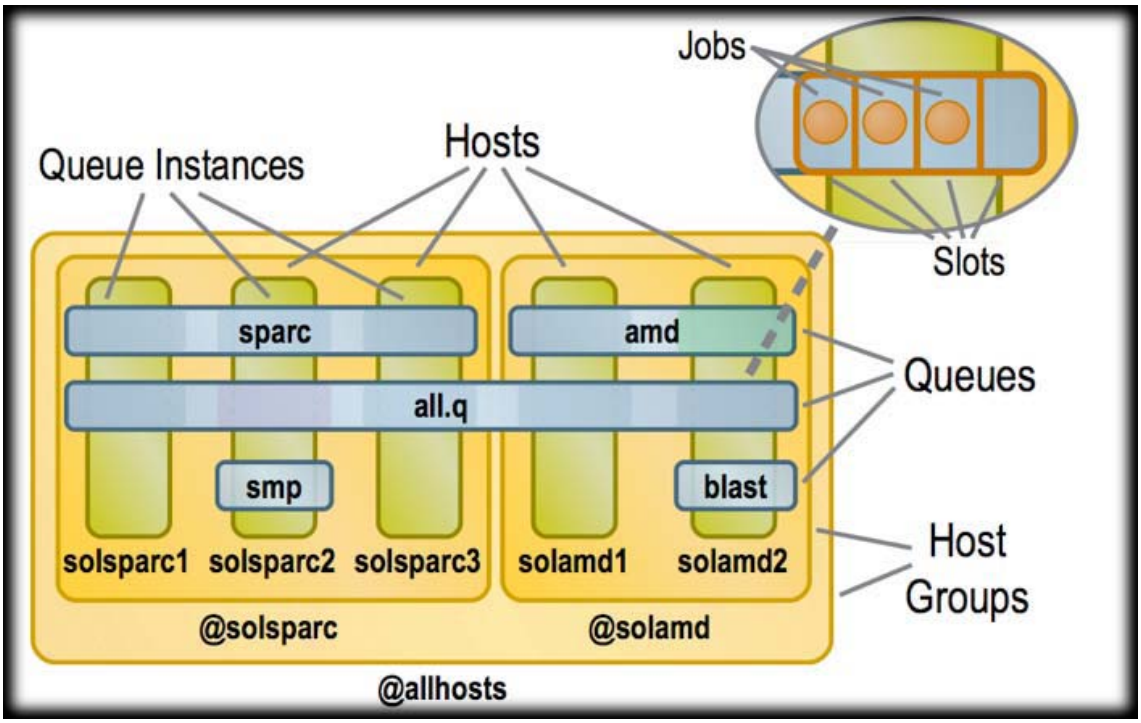
Il·lustració 1 Hosts types & SGE Daemons www.bioteam.net

Funcionament del sistema de cues

Job: Són aquelles sol·licituds de treball (jobs) fetes pels usuaris per tal que el sistema els executi i retorni els resultats. L'usuari no ha de preocupar-se sobre les cues o tipologia de màquines. Tant sols ha de descriure els recursos necessaris per a que el job s'executi correctament mitjançant la comanda `qsub`. El sistema de cues s'encarregarà d'ubicar el procés de manera òptima.

Les tipologies de treball més habituals són:

- Batch
- Interactives
- Parallel
- Checkpoint



Il·lustració 2 Queues, Slots & Host. By Dan Templeton, www.bioteam.net

Queue: o cues són les definicions de les diferents agrupacions a nivell de clúster, segons tipologia i/o necessitats, on els treballs son executats. En l'exemple, les cues estan definides per tipologia de cpu, *sparc* i *amd*, per clúster `all.q` i per necessitats concretes (tipus de memòria i/o disc i/o tipologia de job) `smp` i *blast*.

Paràmetres de configuració:

- Tipologia del Job
- CPU
- Directives d'accés (Usuari, Departament o Grup)
- Diferents entorns concrets com
    - Parallel Environments
    - Checkpointing

Queue Instance: o instàncies de cua són els contenidors pels jobs que s'executen a un host concret, segons uns certs paràmetres de configuració com:

- Tipologia del Job
- CPU
- Directives d'accés (Usuari, Departament o Grup)
- Diferents entorns concrets com
    - Parallel Environments
    - Checkpointing

Slot: Les cues disposen d'un nombre determinat d'espais per executar jobs, per defecte segons el recursos de CPU disponibles.

Host Groups: En la configuració del clúster podem agrupar grups de nodes sota host groups això és interessant quan tenim un cluster heterogeni. Les agrupacions les podem fer, per exemple, per:

- Arquitectura Processador
- Sistema Operatiu
- Arquitectura de Xarxa

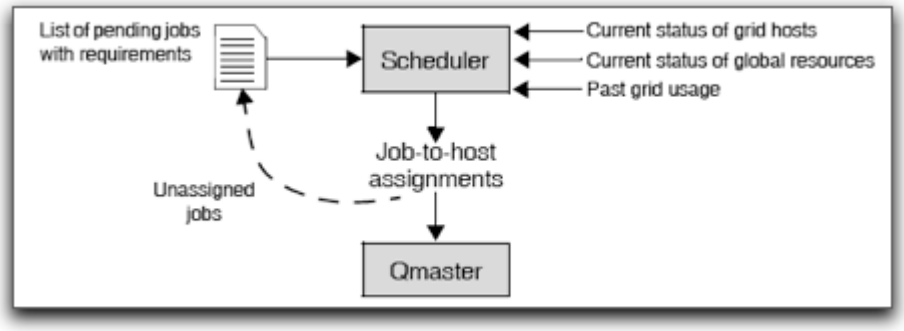
És interessant tenir present que podem realitzar agrupacions de host groups entre host groups.

Scheduler

El planificador scheduler gestiona on i quan els jobs són enviats per a la seva execució.

Les seves funcions són:

- Recol·lectar tota la informació relativa als jobs en execució
- Recol·lectar tota la informació dels hosts mitjançant els `daemons sge_execd`.
- Comprovar l'estat general de tots els recursos del clúster.
- Comprovar l'històric d'ús del sistema.
- Filtrar aquells jobs executables
- Prioritzar els jobs executables.
- Allotjar els jobs en els slots adequats.



Il·lustració 3 Scheduler Policies for Job Priorization in Sun N1GE www.sun.es

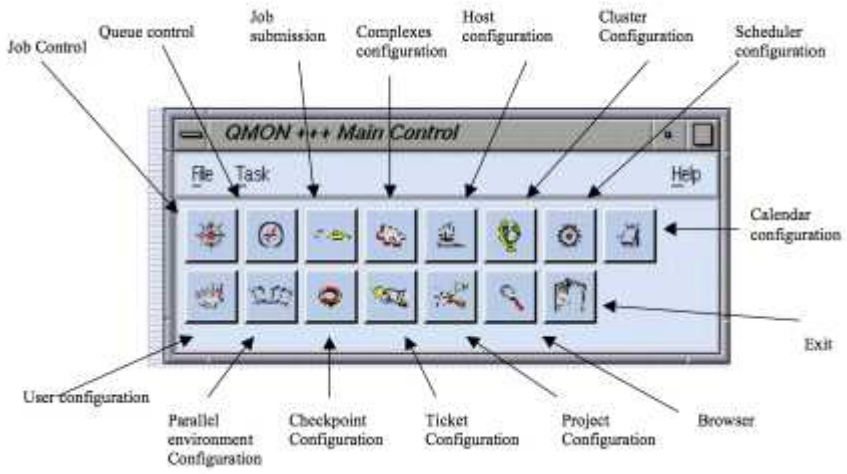
Comandes Bàsiques

`qconf` : Comanda d'administració per afegir, configurar i canviar el sistema de cues.

`qsub` : Comanda d'usuari per enviar jobs al gestor

`qstat` : Comanda per monitoritzar els jobs.

`qmon` [4] : Interfície gràfica tant a nivell d'usuari com d'administració.



[4] https://hec.wiki.leeds.ac.uk/bin/view/Documentation/SgeQmon

Configuració Bàsica

Configuració Execution Hosts

Afegirem tots els nodes clients (màquines virtuals de càlcul) al sistema de cues. És important recordar que és necessari que aquestes estiguin corrent.

Per afegir els nodes un a un des de GE master `qconf -ah hostname`

Altres comandes útils seran:

- `qconf -sel hostname` per mostrar tots els execution hosts
- `qconf -me hostname` modificar un execution host
- `qconf -de hostname` per eliminar el rol d'execution host
- `qconf -se hostname` per mostrar la configuració d'un execution host

Configuració de Host Groups

Crearem un host group per cada tipus d'arquitectura i/o aula per tal de tenir els nodes clients agrupats.

- `qconf -ahgrp @<name>` per a crear el host group
- `qconf -mhgrp @<name>` per a modificar el host group
- `qconf -shgrpl` llistar tots els hostgroups
- `qconf -shgrp @<name>` per mostrar un hostgroup

Configuració General de Cues

`processors`: Atribut per definir el tipus de processador per defecte de la cua. `undefined` per defecte.

`tmp`: Directori temporal. (Normalment `/tmp` or `/scratch`)

`rerun jobs`: Política per defecte d'arracada de jobs que han estat abortats per diferents causes.

`notify time`: temps d'espera per executar certes senyals al sistema com `run`, `suspend` o `kill`.

`slots`: Nombre de jobs que podem concórrer simultàniament a la cua.

`type`: Tipus de cua Batch, Interactiva o ambdues.

`limits`: Limitacions de tipus maquinari o programari com cputime, nombre de cores, quantitat de disc o memòria.

Execució de Jobs

A l'hora d'enviar els jobs a execució haurem de tenir en compte diversos aspectes com els modules o variables a cridar per a la correcta execució d'aquests.

Per això caldrà tenir en compte:

- Prolog: Script que s'executarà abans de l'inici del job
- Epilog: Script que s'executarà després de la finalització del job

També tenim altres opcions com: Starter Method, Suspend Method, Resume Method, Terminate Method.

User Acces

Dins del sistema tenim diferents perfils d'usuari segons les necessitats:

- Managers: Tenen control sobre el sistema
- Operators: Tenen permís sobre el sistema en tots els aspectes menys en aquells relacionats a les cues, és a dir, no tenen capacitat per modificar, afegir o eliminar cues.
- Owners: L'invers dels operadors; poden suspendre, activar i desactivar aquelles cues de les quals són propietaris.
- Users: És el perfil sobre el que treballen els usuaris, tenen accés a totes les consultes però no tenen cap permís de modificació de cap part del sistema.

Grups d'Usuaris

Dins del sistema podem agrupar el usuaris segons tres classificacions o usersets:

- Acces List: Poden existir diferents acces list per diferents recursos. Cada usuari pot pertànyer a diferents acces list.
- Deparment: Poden existir diferents departaments però un usuari només pot pertànyer a un Deparment.
- Project: Poden existir diferents objectes del tipus project però un usuari pot pertànyer a més d'un projecte.

Configuracions complexes OGS

Execution Host

Dins de l'entorn del Execution Host també tenim un conjunt de configuracions complexes de les quals destacarem la variable del `USAGE_SCALING` i `COMPLEX_VALUES`.

- `USAGE_SCALING`: És molt útil en el cas de conviure diferents arquitectures dins del grid. Per exemple, suposem dos aules amb processadors diferents on una té un processador 3 vegades més ràpid que l'altra. Amb aquesta variable podem ponderar l'ús global del sistema en base a la qualitat del recurs.

```
usage_scaling        cpu=9.0
```

- `COMPLEX_VALUES`: En cas de córrer diverses instàncies de cua sobre un mateix node, serà necessari fixar el nombre de slots del execution host com un valor complex.

Configuració de Cues

Checkpointing

Alguns sistemes de cues permeten l'opció de crear checkpoints per tal de, en cas de caiguda del sistema, reprendre els jobs des d'un estat posterior a l'inicial. Això per una banda s'ha de definir en el job i per altra banda ser suportat per aquest.

Les variables més importants són:

- `MinCpuTime` : Interval de temps entre cada checkpoint
- Referenced Ckpt Object : Llista dels enviroments que suporten chekpoint en aquesta cua.

En el cas de OGS cal integrar el mòdul desenvolupat per la universitat de Berkeley BLCR[5] (Berkeley Lab Checkpoint / Restart).

PE

Parallel Environment

Grid Engine permet la possibilitat de proveir als usuaris diferent entorns paral·lels els quals ens donen la possibilitat d'executar jobs paral·lelitzats.

Alguns dels diferents entorns són:

- Programes paral·lels sobre SMP (OpenMp, posix threads, etc...)
- Message-Passing envionments (MPI)

[5] Hi ha disponible els manuals i documentació a: https://ftg.lbl.gov/projects/CheckpointRestart/

És molt recomanable crear Parallel Environments genèrics i homogenis per a totes les cues.

- `qconf -ap <PE name>` per a crear el PE
- `qconf -mp <PE name>` per a modificar el PE
- `qconf –spl` llistar tots els PE
- `qconf -sp <PE name>` per mostrar la configuració d'un PE

Allocation Rules

Dins de la configuració pròpia del PE hem de definir les regles amb les quals s'ubicaran els diferents processos paral·lelitzats entre els execution hosts.

- `$pe_slots`: Amb aquesta configuració tots els jobs seran allotjats dins d'un sol host.
- `$fill_up`: Començant pel host i cua més escaient, els slots disponibles s'aniran omplint sota demanda.
- `$round_robin`: Es realitza una reserva d'un slot a tots els host escaients fins que totes les taques del job estiguin enllestides. Si hi ha més tasques que nodes, aquestes estaran en espera fins que s'alliberi el primer slot.

Exemple de Configuració MPI: `qconf –sp MPI`

```
pe_name mpi
slots 999
user_lists @allusers
xuser_lists NONE
start_proc_args /GE/mpi/startmpi.sh -catch_rsh $pe_hostfile
stop_proc_args /GE/mpi/stopmpi.sh
allocation_rule $fill_up
control_slaves TRUE
job_is_first_task FALSE
urgency_slots min
accounting_summary FALSE
```

Load / Suspend Thersholds

En cas de sobrecàrrega de jobs per part dels usuaris la configuració d'aquestes variables prevenen la caiguda del sistema:

- Load Thresholds: En cas de sobrepassar el llindar de càrrega de jobs, el sistema impedeix que la cua continuï rebent jobs.

- Suspend Thresholds: En cas de sobrepassar el llindar de càrrega de jobs, el sistema suspèn jobs en execució per reduir la càrrega.

Límits i cues subordinades

Els límits ens seran molt útils per crear diferents tipus de cues segons les nostres necessitats o demandes concretes. Així com prevenir abusos dels usuaris o jobs mal dissenyats optimitzant així els recursos.

	Cores (#)	Mem (GB/CORE)	Disk (GB)	cputime(hora)
Fast	-	2 GB	-	2 h
Large	8	2 GB	-	168 h (1 setmana)
Mem	-	4 GB	100 GB	72 h
Disk	-	-	400 GB	72 h

Complementàriament podem fer ús de la subordinació de cues per tal de prioritzar unes cues vers altres.

- Les cues subordinades són suspeses si la cua principal té ocupació plena.
- Les cues tornen a ser disponibles quan la cua principal ja no té ocupació plena.
- Les cues subordinades funcionen mitjançant arbres de dependència.

Complex Resource Attributes

Hi ha tot un seguit de configuracions molt més complexes per tal d'optmitzar i perfilar molt el rendiment del sistema. No mostrarem aquests atributs ja que serien propis d'un treball específic sobre Grid Engine.

Muntatge de Xarxa

El muntatge de xarxa esdevé un element crític en el sistema, sobretot considerant que es treballa sobre una infraestructura concreta sobre la que corren un seguit de serveis els quals son crítics pel funcionament de les aules de docència.

En aquest aspecte tenim diferents possibilitats des de la completa duplicació de la xarxa i els seus elements fins a solucions basades en interfaces virtuals de xarxa.

Virtual Network Interface (VIF)

Una Virtual Network Interface és una representació abstracta i virtual d'una targeta de xarxa la qual pot o no correspondre directament a una targeta de xarxa física.

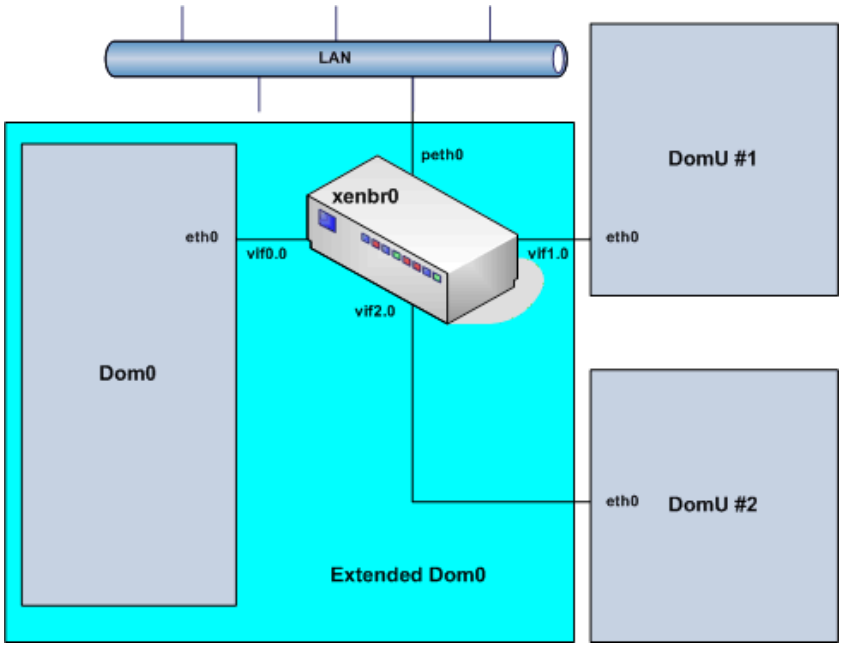
El sistema operatiu segmenta el tràfic generat i rebut per la targeta de xarxa física a través d'un conjunt de targetes de xarxa lògiques.

Virtualització xarxa amb XEN

L'element clau per virtualitzar i aconseguir gestionar les instàncies lògiques de xarxa de cada dom a xen és el Xen Bridge[6].

Sota el domini privilegiat (DomU), corre el `xenbr0` per defecte. Sota aquest bridge virtual, corren totes les vif interface amb nomenclatures `vifx.y` on la X és la representació numèrica del domini i la I és la representació de la interfície al bridge. Aquestes NIC virtuals estan connectades a una de les vif interfaces fent així possible mitjançant el bridging la possibilitat de connexió de les màquines virtuals.

El sistema permet l'ús de diferents targetes físiques així com la creació de nous bridges per segmentar i gestionar el tràfic d'aquestes.



[6] http://wiki.xensource.com/xenwiki/XenNetworking

Serveis LDAP i d'Usuari

El sistema per si mateix no interferirà en sistemes LDAP i d'usuari ja que en assegurar una IP pública a les màquines de docència els sistemes seran totalment compatibles.

Serveis Gestió d'Usuari

Els tres serveis de gestió d'usuaris i maquinari tampoc es veuran afectats per la virtualització de la màquina de docència ja que, tal i com hem mencionat anteriorment, les imatges poden ser distribuïdes directament sobre la màquina virtual.

Els sistemes de control d'usuari com PGINA o el DeepFreeze permeten funcionar directament sobre els sistemes windows de les màquines de docència sense interferir ja que el primer té rutes directes al servidor i xarxa UB mentre que el segon funciona directament sobre el sistema operatiu Windows.

Proposta Tècnica

Introducció

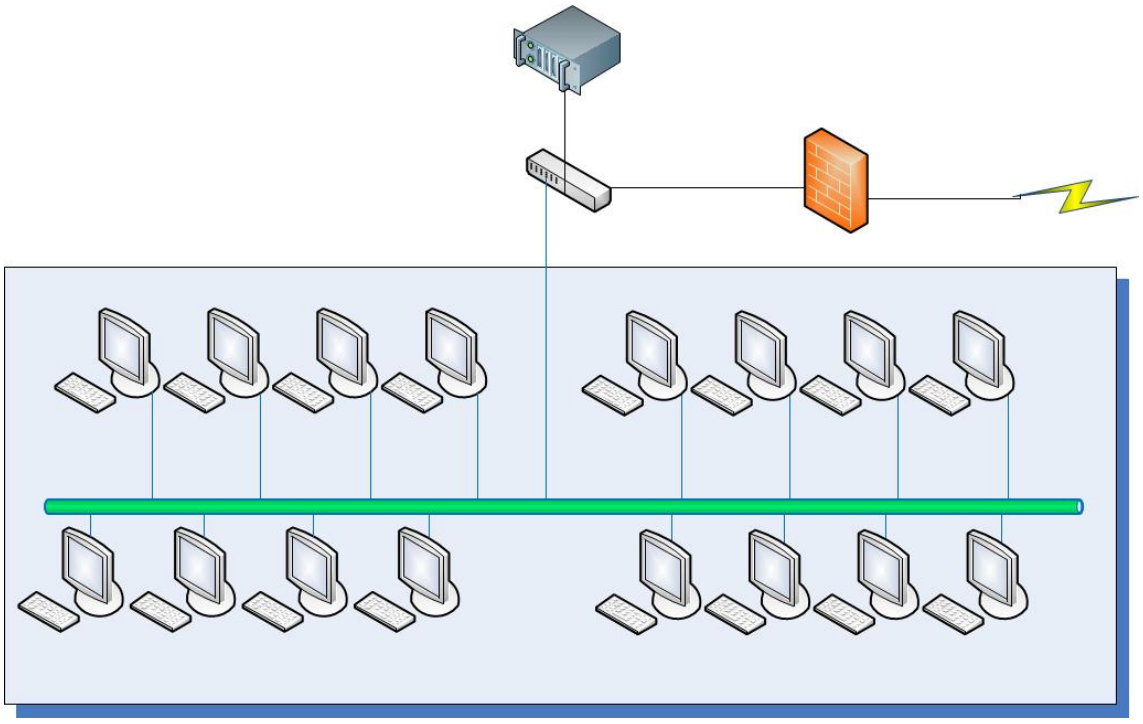
Aquesta proposta és la més senzilla a nivell tècnic i fa un gran èmfasi en la relació cost-eficiència, disminuint així gran part dels costos en maquinari i manteniment.

No per això serà una solució de perfil baix i deficient, sinó el contrari, és una solució que complirà amb totes les funcionalitats desitjades sense interferir en el bon funcionament de l'aula.

El desavantatge estarà en el tipus de configuració STAND-ALONE on des del punt de vista del clúster de càlcul existiran una sèrie de punts de fallida els quals no seran redundant (SPOFS) tant a nivell de maquinari com de programari.

En qualsevol cas s'haurà de garantir el funcionament dels equips de l'alumnat per a poder fer-ne ús sigui quin sigui l'estat de la infraestructura de càlcul.

Aquesta proposta també és molt menys escalable, ja que no permet de cap forma un creixement sostingut. A favor té un cost molt més contingut i un funcionament prou sòlid.



Infraestructura de Servidors

Maquinari

En aquest muntatge tota la infraestructura de càlcul dependrà d'un sol Node que disposarà de tots els recursos de maquinari necessaris per al funcionament del clúster de càlcul.

La redundància en els elements del propi servidor serà clau per minimitzar els punts de fallida i assegurar la integritat de les dades.

El servidor ha de complir els següents requeriments:

- Placa de, com a mínim, dos vies
- Processador Intel Xeon o AMD Opteron amb 4 cores per CPU
- 2GB de memòria cau per core
- Controladora RAID d'entre 4 a 6 terabytes d'espai brut i amb possibilitat de canvi de discos en calent (Hot-Plug)
- Discs durs de rendiment SAS o Sata Near-Line SAS
- Fonts d'alimentació redundades i amb possibilitat de canvi en calent (Hot-Swap)

Programari

El llistat de programari instal·lat en aquest node primari serà (no contemplem tots aquells paquets secundaris necessaris pel funcionament d'ells mateixos):

- COMPILADORS
    - GCC, g77, ifort, icc, porland...
- MODULES
- SUN GRIN ENGINE (Gestor de Cues + Scheduler)
- GANGLIA (Monitorització)
- Biblioteques de Parl·lelització
    - MPI, MPICH
- Biblioteques Matemàtiques
    - BLAS, LIBGOTO, ATLAS

- Aplicacions de Càlcul Optimitzades
- Distributed Shell (permet rèplica de comandes ubicades al fitxer `/etc/dsh/machines.list`)

Des d'aquesta màquina i mitjançant la comanda `qsub` del gestor de cues els usuaris podran llençar els càlculs a la cua de càlcul. Així com comprovar l'estat de tots els seus càlculs.

Estructura Gestor Cues dins de l'entorn del projecte

Dins del nostre projecte amb configuració actual la instal·lació del OGS es farà directament sobre la màquina servidor i seguirem la següent distribució de rols:

- Master Host: Servidor
- Submit Host: Servidor
- Admin Host: Servidor
- Execution Host: Màquines Virtuals de Càlcul

Caldrà tenir en compte que la paral·lelització està limitada al core d'un mateix host, això és degut a la gestió de la infraestructura, ja que en fer un `suspend` dels execution host no podrem garantir que els paquets dels entorns paral·lels que estan circulant en aquell moment per la xarxa siguin desats. Aquesta circumstància produiria un percentatge de jobs fallits molt elevat. Per aquest motiu en la cua que generem la variable `allocation` serà igual a: `$pe_slots`.

Per altra banda, també degut a que la funció de clúster de càlcul només funcionarà fora de l'horari de docència no té cap sentit crear i reservar nodes per a cues batch on el usuaris treballin en temps real.

Per tant, la definició de cues es basarà en una única cua per l'aula i en cas de ser possible la integració, a més, caldria crear un cua per cada tipologia de computador amb diferent arquitectura.

Estructura d'usuari i accés

En aquesta solució l'accés d'usuari es farà per ssh sobre el mateix servidor el qual serà l'encarregat també de gestionar el sistema de fitxers tant amb les dades relatives a les configuracions, com les dades pròpies dels usuaris incloent aquelles generades per el sistema.

Infraestructura d'Aula

Muntatge de Xarxa

El muntatge de Xarxa seguint la línia mestra de la proposta tècnica també respondrà al principi de cost-eficiència destacat anteriorment.

Tot i que, en principi, la solució òptima requeriria d'una xarxa duplicada tant per qüestions de robustesa com d'optimització d'ample de banda, s'ha optat finalment per una solució on s'aprofitaran al màxim les possibilitats de virtualització de Xen a nivell de Xarxa mitjançant l'ús de les Virtual Interface (VIF) i de les capacitats de routing dels switch layer3.

Amb aquesta òptica reduïm la redundància en els elements de xarxa obtenint un decrement econòmic important en el cost global de la solució sense comprometre l'eficiència i en detriment de la robustesa global de la solució.

Muntatge Final

Per tal de mantenir la configuració senzilla i pràctica, la solució més senzilla passa per duplicar el nombre d'IPs públiques assignant una IP pública als DomU, la qual cosa permetria facilitar les connexions en cas de falla de la infraestructura de càlcul i una altra IP per a les dues altres màquines de càlcul.

En garantir una IP UB en la màquina d'usuari assegurem també la possibilitat de traçar rutes cap el servidor SAMBA així com altres serveis de log in de la UB o servidors de llicències.

Per altre banda, mantenint el DomU amb una IP pública simplifiquem la gestió de xarxa del Switch ja que no caldrà duplicar VLANS als diferents ports sinó amb una sola podrà dirigir tot el tràfic.

Si el nombre d'IPs fos crític en la elecció de la solució, es podria solucionar amb la possibilitat d'implementar la xarxa de càlcul amb un rang d'IPs privats per l'aula.

Per a dur a terme aquesta segona opció amb garanties el switch haurà de ser capaç d'enrutar 2 VLANS sobre els mateixos ports per tal de permetre tant a la xarxa privada de càlcul com a la xarxa UB (màquina d'usuari) el correcte tràfic de dades.

La xarxa quedaria estructurada d'aquesta forma amb accés a les dues VLAN.

```
VLAN1 10.0.10.0/24
VLAN2 161.116.0.0/16
      PC0
            domU 10.0.10.10
            Màquina d'usuari 161.116.XXX.XXX
            Màquina de Càlcul 10.0.10.100
      PC1
            domU 10.0.10.11
            Màquina d'usuari 161.116.XXX.XXX
            Màquina de Càlcul 10.0.10.101
      PC2
            domU 10.0.10.12
            Màquina d'usuari 161.116.XXX.XXX
            Màquina de Càlcul 10.0.10.102
```

Distribució del pool d'IPs

Cal tenir en compte que per ambdues solucions caldrà tenir constantment actualitzat l'inventari de màquines de la infraestructura de càlcul (servidor i màquines virtuals) directament sobre el fitxer `/etc/host` on hi haurà un inventari complet de les màquines físiques i virtual així com de les IP del switch.

Això es deu a l'ús del ssh i altres eines de paral·lelització, així com el servei gestor de cues que fan ús d'aquests fitxers per ubicar tant els recursos com per obtenir l'accés a ells.

Exemple /etc/host

```
127.0.0.1 localhost
#DomU Machines
10.0.10.10      pc01 (domU)
10.0.10.11      pc02 (domU)
10.0.10.12      pc03 (domU)
#Calc Machines
10.0.10.100     pc01cal01
10.0.10.101     pc02cal02
10.0.10.102     pc03cal03
```

Això no entra en conflicte amb els servidors DHCP dels serveis de la UB ni de la xarxa de càlcul sempre i quan es tingui la precaució de fixar la IP per a cada màquina mitjançant el hostname.

El principal avantatge que trobem en treballar amb un rang d'IPs completament UB és que la connectivitat cap a l'exterior sempre estarà garantida tant des del DomU com des de les màquines. Per contra, farà tediosa la tasca de canviar els rangs d'IPs en cas de necessitat.

En canvi, l'ús de rangs privats, tot i augmentar la complexitat de la configuració inicial de la solució per tal d'assegurar la sortida a l'exterior des de la infraestructura de càlcul, permetria més independència sobre els rangs propis de la UB, ja que serien transparents a la infraestructura de càlcul.

Muntatge Clients

El muntatge dels clients passa per la solució Xen vista anteriorment sense canvis ni modificacions.

Anàlisi de Costos

En l'anàlisi de costos partirem de les condicions inicials d'una aula on només hi ha la infraestructura de comunicacions bàsica instal·lada.

A partir d'aquesta base valorarem el cost relatiu a la inversió realitzada per adquirir tot l'equipament necessari pel funcionament de la solució, excloent la partida econòmica relativa a les hores home necessàries per instal·lar i posar-la en marxa.

Resum Costos[7]				
Infraestructura de Servidors				
Dell PowerEdge R510[8]	8 Cores / 16Gb Ram/ 6Tb HDD	1	6.856,00 €	6.856,00 €
Rack		1	649,00 €	649,00 €
				7.505,00 €
Infraestructura d'Aula				
Estació de Treball[9]		20	960,00 €	19.200,00 €
				19.200,00 €
				26.705,00 €

Aquesta proposta serà, doncs, ideal per desenvolupar un sistema de demostració i avaluació de la solució.

[7] No s'inclouen les partides relatives a impostos
[8] Preu orientatiu segons mercat www.dell.es
[9] Preu mig per estació de treball estimat segons licitació Universitat de Barcelona Expedient 2011/42

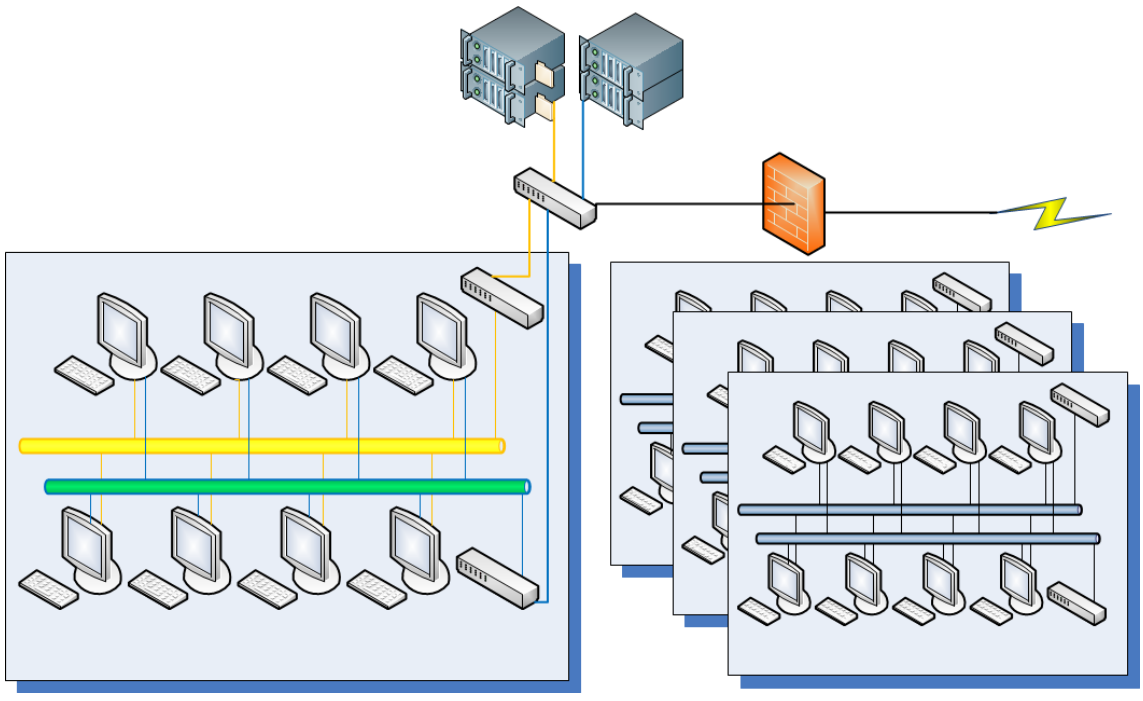
Proposta ideal

Introducció

Aquesta solució engloba una infraestructura molt més sòlida, eficient i escalable a tots el nivells.

Entre les principals diferències que podem trobar apareixen una doble infraestructura de servidors redundada plenament tant a nivell de maquinari com de programari, una duplicació de la xarxa interna de l'aula, així com la possibilitat d'incloure sota el paraigües la mateixa infraestructura amb garanties vàries aules dins del mateix entorn de la facultat.

El nivell de profunditat d'aquesta solució serà molt menor ja que podríem considerar cada una de les especialitats com projectes individuals. En el món empresarial on aquest tipus de solucions són comuns en la part aliena al HPC. Els projectes són tractats per diversos consultors especialitzats a nivell de servidors virtuals, dispositius d'emmagatzematge i infraestructures de xarxa.



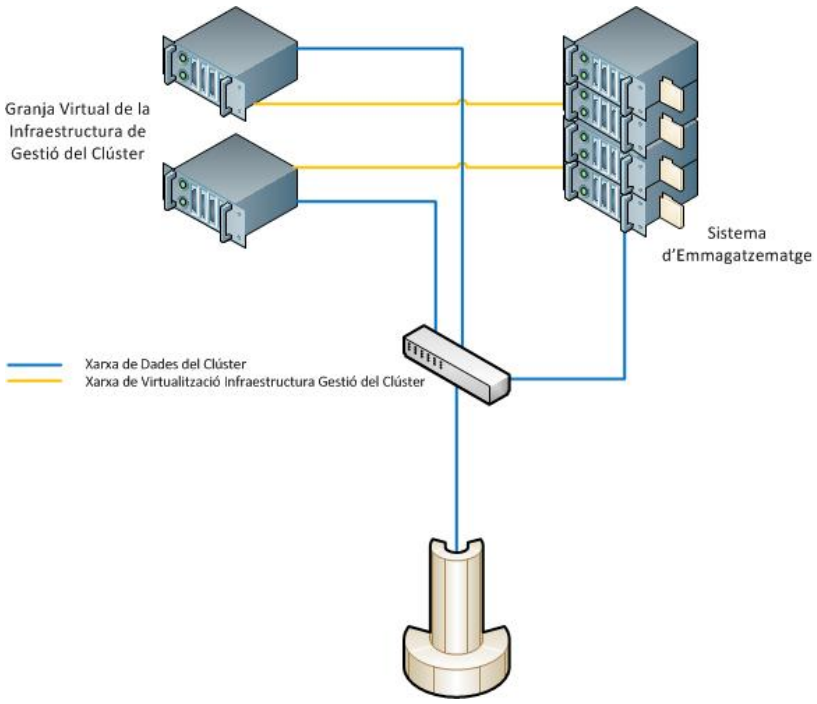
Infraestructura de Servidors

Maquinari

En aquest muntatge tota la infraestructura estarà redundada a nivell de servidors de prestacions equivalents de la solució anterior però amb la diferencia que en aquesta solució comptarà amb un servidor de fitxers SAN (*Storage-Area-Network*) amb doble controladora traslladant la capacitat de disc i la gestió d'aquesta solució d'emmagatzemament.

Això ens permetrà treballar amb màquines virtuals a nivell de servidors permetent balancejar la càrrega entre els servidors físics i garantint la possibilitat d'un creixement sostingut de recursos.

Els requeriments propis de la SAN no estan inclosos degut a la gran diversitat de solucions que podem trobar al mercat. No obstant apuntarem com condició molt important el requeriment de que sigui operable amb el protocol iSCSI, ja que farà compatible la solució amb la infraestructura de xarxa ethernet de la solució sense haver de passar per solucions de fibra òptica.



La SAN serà l'encarregada de, per una banda servir als servidors mitjançant xarxa les dades per córrer els sistemes virtuals, i de l'altra servir de sistema d'emmagatzemament pels usuaris.

En aquesta solució no entrarem a valorar els paquets o requeriments de cada sistema com hem fet anteriorment sinó que veurem tot allò necessari per poder implementar la infraestructura.

Granja de Servidors Virtuals

A nivell de granja, els equips hauran de comptar amb solució de virtualització sent possible des d'una implementació directament amb XEN com un desplegament de sistemes comercials com VMWare, Citrix o HyperV.

En aquest cas cal valorar seriosament tot el que les solucions comercials poden aportar a nivell de manteniment, seguretat i suport vers una solució directament XEN implementada sense el suport de cap proveïdor de hardware o software.

Els nodes servidors només correran el sistema operatiu del sistema de virtualització, ja que els sistemes propis de la solució estaran dins de la infraestructura virtual.

Podrem generar tot un seguit de màquines virtuals per tal de securitzar cada un dels elements claus del nostre sistema, els quals es faran seguint desplegaments semblants a la solució anterior.

Els servidors virtuals que desplegarem seran:

- Servidor d'Accés d'usuaris – `Portal`
- Servidor d'administració, recull logs i control – `AdminSrv`
- Servidor del Gestor Cues – `MasterOGS`
- Servidor del Gestor de Cues Secundari – `ShadowOGS`

La infraestructura també podrà créixer en màquines virtuals per tal d'oferir nous serveis. A més d'oferir alta disponibilitat en tots aquests serveis a nivell d'infraestructura. Si un dels nodes de virtualització cau degut a un mal funcionalment del hardware, gràcies a la SAN i a la compartició de l'emmagatzematge, les màquines virtuals d'aquest migrarien automàticament al node disponible.

Estructura Gestor Cues dins de l'entorn del projecte

Cada rol serà desenvolupat per una o més màquines virtuals

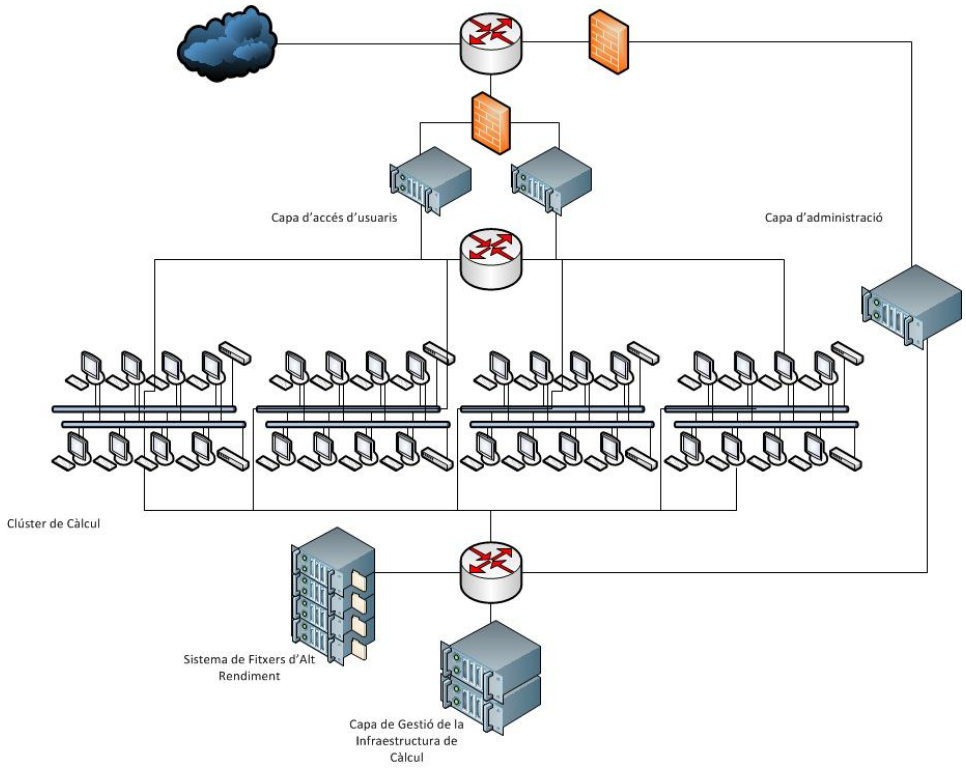
- Master Host: `MasterOGS`
- Shadow Master Host: `ShadowOGS`
- Submit Host: `Portal`
- Admin Host: `AdminSrv`
- Execution Host: Màquines Virtuals de Càlcul

Muntatge de Xarxa

Caldrà el disseny d'una infraestructura de xarxa pròpia per a la infraestructura de servidors i de càlcul connectada també a la xarxa de la UB.

Aquest muntatge complex permetrà l'arribada de trànsit entre les diverses xarxes de dades d'emmagatzematge i de la universitat per tal de garantir tots els serveis.

Cal aclarir també que tant la capa d'accés d'usuaris com la capa d'administració són màquines virtuals allotjades dins de la capa de gestió de la infraestructura de càlcul.



Infraestructura d'Aula

Muntatge de Xarxa

Tenint en compte la xarxa de la UB, caldrà duplicar les boques de l'aula amb un switch que caldrà administrar i permetre tant la creació de VLANS com l'administració de Layer 3.

Aquest permetent crear diferents perfil, així doncs es podria segmentar aquesta administració fent possible que els responsables del clúster només accedissin a la xarxa de càlcul, evitant interferències amb la UB.

Això també ens permetrà millorar el rendiment del sistema en els càlculs amb entorns paral·lels ja que alliberarà de trànsit la xarxa de dades, ja que el trànsit de les xarxes d'accés i emmagatzematge correrà en la segona.

Totes les estacions de treball inclouran dos NIC independents, amb dues IP diferents. La IP de la xarxa de càlcul i administració vindrà donada pel node servidor, que assegurarà el funcionament del clúster. Tot el tràfic generat pels nodes de càlcul del clúster sortirà a través d'aquest per poder monitoritzar l'ús.

Aquesta segmentació assegura que en cas de caiguda del servei DHCP de la xarxa de càlcul o del propi servidor, sigui possible continuar fent un ús acadèmic de l'aula, ja que les estacions de treball i les màquines virtuals de docència seguiran conservant la IP subministrada pel DHCP de la Universitat.

Muntatge Clients

El muntatge del clients passa per la solució xen vista anteriorment sense canvis ni modificacions.

Anàlisi de Costos

En l'anàlisi de costos partirem de les condicions inicials d'una aula on només hi ha la infraestructura de comunicacions bàsica instal·lada.

A partir d'aquesta base valorarem el cost relatiu a la inversió realitzada per adquirir tot l'equipament necessari pel funcionament de la solució, excloent la partida econòmica relativa a les hores home necessàries per instal·lar i posar en marxa aquesta.

Resum Costos[10]				
Infraestructura de Servidors				
Dell PowerEdge R310[11]	4 Cores / 8GBRam /600GB HDD	2	1.749,00 €	3.498,00 €
Equalogic PS4110[11]	6Tb HDD	1	19.000,00 €	19.000,00 €
Rack		1	649,00 €	649,00 €
				23.147,00 €
Infraestructura d'Aula				
Estació de Treball[12]		20	960,00 €	19.200,00 €
Targeta NIC Addicional[13]		20	17,75 €	355,00 €
Cablejat Sala (Tarifa UB)[13]		20	222,96 €	4.459,20 €
				24.014,20 €
				47.161,20 €

Tot i que d'entrada és una solució molt cara i per tant difícil de justificar i dur a terme, cal pensar també que és la infraestructura de gestió bàsica per a qualsevol infraestructura de càlcul amb garanties i rendiment necessari per administrar recursos de càlcul d'altes prestacions.

Caldria pensar, doncs, en que a mig i llarg termini aquesta solució permetria la integració de solucions de càlcul amb més rendiment, a més de l'ús de les aules per càlculs menys crítics i com a primer contacte amb aquestes infraestructures i tot sota una mateixa infraestructura de gestió.

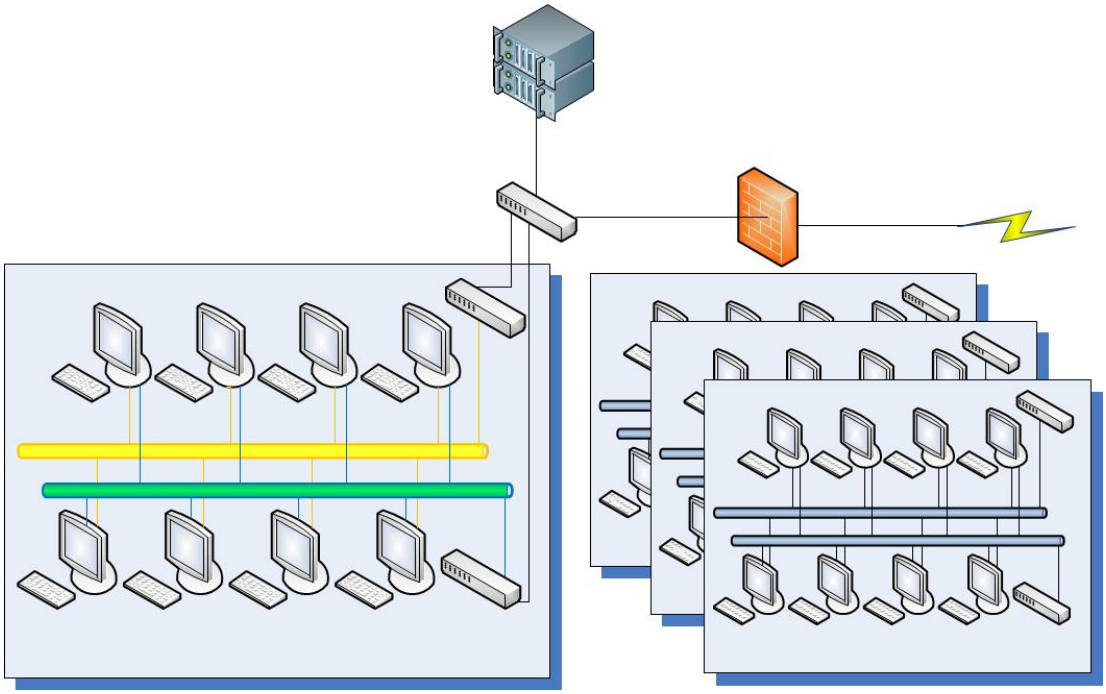
[10] No s'inclouen les partides relatives a impostos
[11] Preu orientatiu segons mercat www.dell.es
[12] Preu mig per estació de treball estimat segons licitació Universitat de Barcelona Expedient 2011/42
[13] Preu segons Sol·licitud de Connexió a La Xarxa Informàtica de la UB - Any 2012 (Inclou l'electrònica de xarxa)

Solucions Intermèdies

Introducció

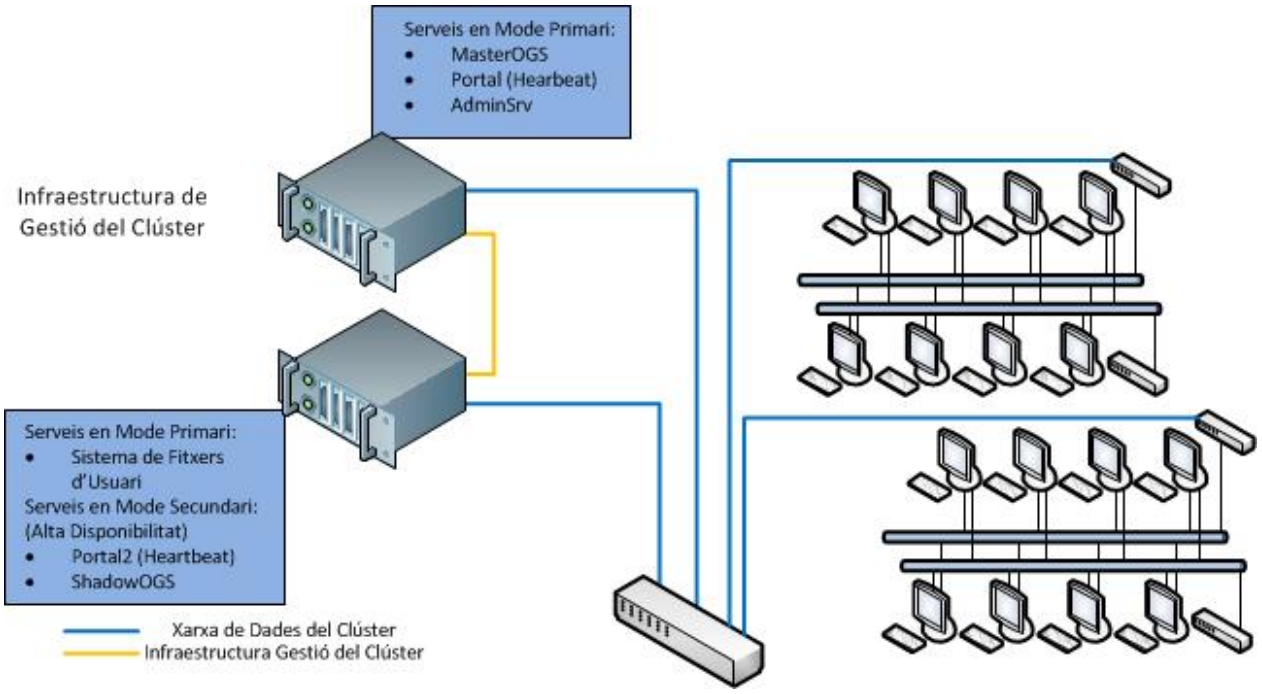
Entre ambdues solucions hi ha la possibilitat d'implementar tot un seguit de solucions intermèdies.

Una d'elles, amb un cost molt més contingut, passaria per eliminar la part d'emmagatzematge de la proposta ideal. Això permetria reduir de forma considerable els costos mantenint, en part, l'alta disponibilitat.



Aquesta infraestructura idèntica a l'anterior en els aspectes d'infraestructura d'aula i infraestructura d'Aula i semblant a nivell de servidors on un servidor tindria el rol d'emmagatzemar tota la infraestructura de fitxers i la còpia dels serveis clau de la infraestructura, i l'altre que assumiria, mitjançant màquines virtuals, tots el serveis clau en mode primari.

Això seria possible amb tecnologies d'alta disponibilitat com la instal·lació de `heartbeat`. Un `daemon` de Linux que permet controlar la presència o desaparició de serveis i màquines.



Anàlisi de Costos

En l'anàlisi de costos partirem de les condicions inicials d'una aula on només hi ha la infraestructura de comunicacions bàsica instal·lada.

A partir d'aquesta base valorarem el cost relatiu a la inversió realitzada per adquirir tot l'equipament necessari pel funcionament de la solució, excloent la partida econòmica relativa a les hores home necessàries per instal·lar i posar en marxa aquesta.

Resum Costos[14]				
Infraestructura de Servidors				
Dell PowerEdge R510[15]	8 Cores / 16Gb Ram/ 6Tb HDD	1	6.856,00 €	6.856,00 €
Dell PowerEdge R310[15]	4 Cores / 8GBRam /600GB HDD	1	1.749,00 €	1.749,00 €
Rack		1	649,00 €	649,00 €
				9.254,00 €
Infraestructura d'Aula				
Estació de Treball[16]		20	960,00 €	19.200,00 €
Targeta NIC Addicional[17]		20	17,75 €	355,00 €
Cablejat Sala (Tarifa UB)[17]		20	222,96 €	4.459,20 €
				24.014,20 €
				33.268,20 €

[14] No s'inclouen les partides relatives a impostos
[15] Preu orientatiu segons mercat www.dell.es
[16] Preu mig per estació de treball estimat segons licitació Universitat de Barcelona Expedient 2011/42
[17] Preu segons Sol·licitud de Connexió a La Xarxa Informàtica de la UB - Any 2012 (Inclou l'electrònica de xarxa)

En aquesta solució podríem reduir fins a aproximadament un terç la inversió de la proposta ideal en infraestructura de servidors millorant la relació cost-rendiment en detriment de l'alta disponibilitat, però millorant en tots els aspectes la relació cost-rendiment - alta disponibilitat de la proposta tècnica augmentant la complexitat de tot el sistema.

Això fa, a mig termini, la solució més idònia, ja que permetria l'ampliació del clúster a diverses aules i més fàcilment convertible gràcies a la seva modularitat.

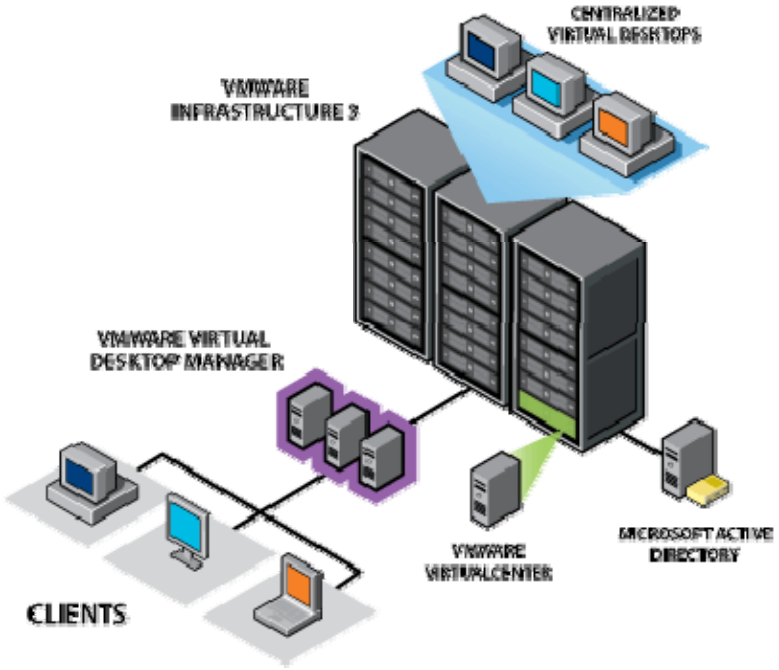
Perquè no somiar?

Des de fa temps la idea thin client i escriptoris remots ha guanyat força en entorns empresarials degut a la reducció del cost d'inversió i sobretot als costos indirectes sobre l'administració dels equips i usuaris.

Virtual Desktop Interface

Tot i que la tecnologia d'escriptoris remots oferts sota demanda està basada en la antiga idea del terminal-servidor, gràcies a la tecnologia de la virtualització ha agafat molta empenta.

El funcionament és senzill i la idea d'agafar els avantatges dels antics terminals i la flexibilitat del sobretaula milloraria la gestió d'administració així com la seguretat i criticitat de la informació.



Imaginem, doncs, la possibilitat de compartir i aprofitar aquests recursos per l'ús HPC a un cost semblant i un rendiment elevat amb la possibilitat de compatibilitzar i balancejar aquest amb els serveis d'usuari d'escriptoris sota demanda.

L'escriptori passaria a ser per l'alumne i pel personal docent i investigador un servei més, amb un valor afegit molt important ja que, a part d'alleugerar l'ocupació de les aules en moments

puntuals, permetria connectar a qualsevol alumne o PDI des de qualsevol màquina per a treballar pràcticament des de qualsevol ubicació amb xarxa amb tots el recursos propis de la universitat.

Per altra banda, també permetria als serveis informàtics de la universitat reduir costos d'administració i gestió així com reduir i sostenir la inversió en maquinari d'una forma molt més eficient i econòmica.

Tot l'escrit anteriorment permetria posar en valor les TIC en un entorn universitari on per desgràcia solen ser més un obstacle que un valor afegit.

BIBLIOGRAFIA

1. *Grid Engine Training'12* [En línia]. Barcelona: HPC Knowledge Portal, Xarxa de Referència en Química Teòrica i Computaciona España, 2012 [Consulta: 25 abril 2012]. Disponible a : <http://www.hpckp.org/index.php/training/grid-engine-training-12>.
2. *Dom0 Kernels for Xen* [En línia]. Fort Lauderdale, Florida: Citrix Systems Inc., 2012 [Consulta: 2 gener 2012]. Disponible a : < http://wiki.xen.org/wiki?title=Dom0_Kernels_for_Xen&action=history >.
3. *Top500 November 2011 List*[En línia]. Mannheim: University of Mannheim, 2011 [Consulta: 3 desembre 2011]. Disponible a : <http://www.top500.org/lists/2011/11>.
4. *Commands List for Managing Xen Virtual Machines from the XenServer Host CLI* [En línia]. Fort Lauderdale, Florida: Citrix Systems Inc. , 2008 [Consulta: 19 gener 2012]. Disponible a : <http://support.citrix.com/article/CTX116016>.
5. *Installing Xen* [En línia]. IPSSystem, 2012 [Consulta: 5 maig 2012]. Disponible a : <http://en.ispdoc.com/index.php/Installing_XEN>.
6. *Xen Howtos* [En línia]. Provo, Utah: Novell Inc., 2011 [Consulta: 21 gener 2012]. Disponible a : <http://en.opensuse.org/openSUSE:SUSE_Studio_howtos#Xen_Howtos>.
7. *Deploying Sun Grid Engine* [En línia]. New Jersey: The Globus Consortium, 2006 [Consulta: 9 maig 2012]. Disponible a : <http://www.globusconsortium.org/tutorial/ch3/page_3.php>.
8. *Open Grid Scheduler* [En línia]. Fremont, California: VA Software, 2009 [Consulta: 26 març 2012]. Disponible a : <http://gridscheduler.sourceforge.net/>.
9. *Open Grid Scheduler Documentation*[En línia]. Fremont, California: VA Software, 2010 [Consulta: 26 març 2012]. Disponible a : < http://gridscheduler.sourceforge.net/documentation.html>.
10. *Grid Engine Administration – Training Slides*[En línia]. Massachusetts: Bioteam Inc.,2009[Consulta: 7 març 2012]. Disponible a : <http://bioteam.net/2009/09/sge-training-slides/>.
11. *Workflow Enviorenments Guide* [En línia]. Massachusetts: Bioteam Inc., 2008 [Consulta: 7 març 2012]. Disponible a : < http://bioteam.net/2005/08/workflow-environments-guide/ >.
12. *Building Open Grid Scheduler on CentOS/RHEL 6.2*[En línia]. Massachusetts: Bioteam Inc., 2012 [Consulta: 8 març 2012]. Disponible a : < http://bioteam.net/2012/01/building-open-grid-scheduler-on-centos-rhel-6-2/ >.
13. *Time Syncronization on Xen Setup*[En línia]. Provo, Utah: Novell Inc., 2009 [Consulta: 22 abril 2012]. Disponible a : < http://www.novell.com/communities/node/8629/time-synchronization-xen-setup >.
14. *N1 Grid Engine Installation Guide*[En línia]. Redwood Shores, California: Oracle Corp., 2010 [Consulta: 17 Maig 2012]. Disponible a : < http://www.novell.com/communities/node/8629/time-synchronization-xen-setup >.
15. *Integrating Grid Engine Workflows to Clouds* [En línia]. Lisle, Illinois, 2011 [Consulta: 26 maig 2012]. Disponible a : < http://www.univa.com/products/unicloud >.

ANNEX I – INSTAL·LACIONS

Instal·lació XEN

Procés d'Instal·lació XEN HYPERVISOR 4.0

Paquets a instal·lar

- vim
- build-essential
- gfortran
- bzip2
- sudo
- sudosh2
- xen-tools
- xen-linux-system-2.6.32-5-xen-amd64
- xen-hypervisor-4.0-amd64
- linux-headers-2.6.32-5-xen-amd64
- linux-headers-2.6.32-5-common-xen
- libxenstore3.0
- lxen-utils-4.0

En aquesta màquina instal·larem tots aquells paquets i serveis que considerem necessaris, així com l'accés a internet i la resta de dispositius del clúster.

Posada en Marxa

Treballarem sota Logical Volume Groups, ja que ens dóna molta flexibilitat alhora d'ampliar, moure i treballar amb aquests volums lògics en comptes de amb particions més tradicionals.

Utilitzarem 3 particions en cada màquina de docència: una per el sistema i el domU, una segona per emmagatzemar les màquines virtuals i una tercera que muntarem només des de les màquines virtuals de càlcul per usar com scratch durant els càlculs.

Dins l'estructura lvm tindrem els següents elements:

- Physical groups:
  - `/dev/sdb`
- Volum groups:
  - `vgsys`
  - `vgvmachines` (contrindrà les màquines virtuals de càlcul i docència)
  - `vgscratch`

- lvm per el Volum group vgvmachines:
  - pc01_calcul
  - pc01_calcul_swap
  - pc01_docencia
  - pc01_doc**e**ncia_temp

Creem `fdisk /dev/sdb`, una segona partició per allotjar les màquines virtuals, la partició de sistema i de scratch ja han estat creades durant la instal·lació:

```
 Device Boot      Start         End      Blocks   Id  System
/dev/sda1               1       15666   125837113+  83  Linux
/dev/sda2           15667       31332   125837145   83  Linux
/dev/sda3           31333       46998   125837145   83  Linux
```

Preparem les particions del disc lliure, per crear els volume groups o discs virtuals.

```
root@pc01:~#pvcreate /dev/sda2
```

```
root@pc01:~# pvdisplay
  --- Physical volume ---
  PV Name               /dev/sda
  VG Name               vgsys
  PV Size               66.27 GiB / not usable 4.00 MiB
  Allocatable           yes
  PE Size               4.00 MiB
  Total PE              16963
  Free PE               12331
  Allocated PE          4632
  PV UUID               TvgvzQ-GC9U-ccfH-zqNt-jeW4-qpmM-LDXZWH

  "/dev/sda2" is a new physical volume of "120.01 GiB"
  --- NEW Physical volume ---
  PV Name               /dev/sda
  VG Name
  PV Size               120.01 GiB
  Allocatable           NO
  PE Size               0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               tfsxne-EGjH-R8hl-Ymw0-Aovb-lZBX-J14kDS
```

Formatgem en format ext3,

```
mkfs.ext3 /dev/sda2
```

Crearem el volume group `vgvmachines`:

```
root@pc01_domU:~# vgcreate vgvmachines /dev/sda2
  No physical volume label read from /dev/sda2
  Physical volume "/dev/sda2" successfully created
  Volume group "vgvmachines" successfully created
```

Crearem totes les particions on s'ubicaran les màquines virtuals:

```
lvcreate -L1.00GB -n pc01_calcul vgvmachines
lvcreate -L10.00GB -n pc01_calcul_swap vgvmachines

lvcreate -L4.00GB -n pc01_docencia vgvmachines
lvcreate -L10.00GB -n pc01_docencia_temp vgvmachines
```

Donem format a les particions,

```
mkfs.ext3 /dev/vgvmachines/pc01_calcul
mkswap /dev/vgvmachines/pc01_calcul_swap


mkfs.fat32 /dev/vgvmachines/pc01_docencia
mkfs.fat32 /dev/vgvmachines/pc01_docencia_temp
```

La creació de les màquines virtual Linux es farà a partir de la creació d'una màquina virtual que usarem com a plantilla i anomenarem `base`.

```
xen-create-image --force  --passwd --dhcp --lvm vgvmachines --hostname base
```

En fer un `lvdisplay`, apareix la nova màquina creada:

```
  --- Logical volume ---
  LV Name                /dev/vgvmachines/base
  VG Name                vgvmachines
  LV UUID                egTFuQ-FOMV-cPMv-Mwj2-VTkE-OeL4-mEC6DG
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                128.00 MiB
  Current LE             32
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           254:8

  --- Logical volume ---
  LV Name                /dev/vgvmachines/base
  VG Name                vgvmachines
  LV UUID                ktepA1-zV25-iyzP-k6JK-63Sd-WOx6-mG5PPL
```

```
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                4.00 GiB
  Current LE             1024
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           254:9
```

Això crea l'arxiu : `/etc/xen/base.cfg`

Per arrancar la màquina virtual usarem la comanda

```
xm create -c /etc/xen/base.cfg
```

En aquesta màquina instal·larem tots aquell paquets i serveis que considerem necessaris, així com l'accés a internet i la resta de dispositius del clúster.

Clonar màquina virtual

Màquina Virtual de Càlcul

Des de `pc01`, farem un snapshot de la màquina virtual base:

```
lvcreate -s -L 1G -n vm.snap_base_root /dev/vgvmachines/base-disk
```

Muntem el snapshot de la base i una de les particions per màquines virtuals que havíem creat. La carpeta vserver s'utilitzarà per cadascuna d'aquestes particions:

```
mkdir -p /mnt/snap_base/
mkdir -p /mnt/vserver/
mount /dev/vgvmachines/vm.snap_base_root /mnt/snap_base/
mount /dev/vgvmachines/sgae_root /mnt/vserver/
```

Esborrem el contingut d'aquesta carpeta perquè ens fa nosa i movem el contingut del snapshot cap a on està muntada la futura màquina virtual.

```
rm -rf /mnt/vserver/lost+found/
rsync -altgvb /mnt/snap_base/*  /mnt/vserver/
```

Un cop fet això, netegem el sistema i crearem i provarem la màquina virtual per veure que funciona correctament abans de seguir amb la resta de màquines virtuals. D'entrada desmuntem les particions i esborrem el lv del snap shot (després s'ha de tornar a crear):

```
umount /mnt/snap_base/
umount /mnt/vserver

 lvremove -f /dev/vgvmachines/vm.snap_base_root
```

Caldrà crear un fitxer de configuració per aquesta màquina virtual. Partim del `base.cfg` i fem les següents modificacions:

```
cd /etc/xen
cp base.cfg pc01_calcul.cfg
vi pc01_calcul.cfg
...
disk        = [
                  'phy:/dev/vgvmachines/pc01_calcul,xvda2,w',
                  'phy:/dev/vgvmachines/pc01_calcul_swap,xvda1,w',
              ]
...
#
#  Hostname
#
name        = 'pc01calcul'
#
#  Networking (La mac ens la invintem)
#
dhcp        = 'off'
vif         = [ 'mac=00:16:3E:A8:41:77,bridge=xenbr1' ]
```

Creem la màquina virtual, entrant directament a consola:

```
xm create -c /etc/xen/pc01_calcul.cfg
```

Un cop a dins de la nova màquina virtual canviem totes aquelles variables comunes a la plantilla base. Així com configurarem tot allò que sigui necessari.

També la inclouríem com a execution host i assegurarem la visibilitat dels directoris del Grid Engine.

Un cop configurada la màquina moure la màquina virtual a un directori intern `/etc/xen/càlcul/` per tal d'evitar l'arrencada d'aquesta amb el sistema. Forçant així l'arrencada manual.

Màquina Virtual de docència

A causa de la falta de llicències propietàries de Microsoft Windows XP no s'ha pogut fer la instal·lació d'un sistema Windows.

El procés d'instal·lació de la màquina Windows es realitzaria començant per la instal·lació d'una màquina Windows dins de xen amb la documentació següent:

- *Installing and running Windows XP or Vista as a Xen Domain HVM domainU Guest* [En línia]. Virtopia, 2009 [Consulta: 29 Maig 2012]. Disponible a : <http://www.virtuatopia.com/index.php/Installing_and_Running_Windows_XP_or_Vista_as_a_Xen_HVM_domainU_Guest >.
- *How to Install Windows XP o Vista on Xen* [En línia]. MediaKey.dk 2007 [Consulta: 29 maig 2012]. Disponible a : <http://mediakey.dk/~cc/howto-install-windows-xp-vista-on-xen/ >.

Un cop instal·lada la màquina Windows bàsica de docència, procedirem a, mitjançant els sistemes propis de la universitat, integrar-la dins del sistema propi de replicació de la universitat o a la instal·lació manual d'un gestor d'arrencada i un sistema Linux.

Instal·lació Grid Engine

La primera consideració serà que partim dels binaris de la última versió lliure de Sun Grid Engine 6.2 update 5, ja que per altres instal·lacions caldria compilar específicament el codi per optimitzar-lo a la arquitectura.

Per a dur a terme una correcta instal·lació caldrà tenir en compte els següents aspectes:

- Es crític que els execution host comparteixin per NFS la carpeta /sge del Master Host
- En tots el nodes s'ha d'inserir la línia `./sge/default/common/settings.sh` a l'arxiu `/etc/profile`
- Cal disposar dels paquets:
    - `nfs-kernel-server nfs-comon`
    - `build-essentials`
    - `libmotif-dev libmotif3 libxpm-dev libxmp4`

Un cop tenim el binaris de la última release a la carpeta /sge, fixarem la variable d'entorn SGE_ROOT, executarem l'instal·lador durant el qual afegirem el node com a submit host i adminitrative host.

```
export SGE_ROOT=/sge
./install_qmaster
source /sge/default/common/settings.sh
```

Creem els host groups del clúster, en el nostre cas, degut a la seva homogeneïtat només en crearem un.

```
qconf -ahgrp @aula1
(a=add, hg=hostgroup, r=replace, p=purge)
```

Creació d'usuaris

```
qconf –au jordijose lab
(a=add, u=user, group=lab)
```

Creació de Cues

```
qconf -sq aula1.q

qname aula1.q                       notify 00:00:60
hostlist @aula1                     owner_list NONE
seq_no 0                            user_lists jb lab
load_thresholds                     xuser_lists NONE
np_load_avg=1.75                    subordinate_list NONE
suspend_thresholds NONE             complex_values exclusive=true
nsuspend 1                          projects NONE
suspend_interval 00:05:00           xprojects NONE
priority 0                          calendar NONE
min_cpu_interval 00:05:00           initial_state default
processors UNDEFINED                s_rt INFINITY
qtype BATCH INTERACTIVE             h_rt INFINITY
ckpt_list NONE                      s_cpu INFINITY
pe_list make smp                    h_cpu INFINITY
rerun FALSE                         s_fsize INFINITY
slots 12                            h_fsize INFINITY
tmpdir /scratch                     s_data INFINITY
shell /bin/bash                     h_data INFINITY
prolog NONE                         s_stack INFINITY
epilog NONE                         h_stack INFINITY
shell_start_mode                    s_core INFINITY
posix_compliant                     h_core INFINITY
starter_method NONE                 s_rss INFINITY
suspend_method NONE                 h_rss INFINITY
resume_method NONE                  s_vmem INFINITY
terminate_method NONE               h_vmem INFINITY
```

Cal tenir en compte crear les cues amb el pe_list make smp (same machine processing) perquè els càlculs no surtin a l'exterior.