Títol:	Disseny i implementació d'un clúster de computació científica basat en rocks
Autor:	José María Heredia Genestar
Institució:	Institut de Biologia Evolutiva
Data:	17 de Juny de 2011
Director:	Arcadi Navarro i Cuartiellas Institut de Biologia Evolutiva
Ponent:	Ramón Canal Corretger Departament d'Arquitectura de Computadors
Titulació:	Enginyeria Informàtica
Centre:	Facultat d'Informàtica de Barcelona (FIB)
Universitat:	Universitat Politècnica de Catalunya (UPC) BarcelonaTech

Índex

Introducció

L'Institut de Biologia Evolutiva (IBE) és un Institut de recerca de recent creació fruit de la unió de la Unitat de Biologia Evolutiva de la Universitat Pompeu Fabra i el Departament de Fisiologia i Biodiversitat Molecular del Centro Superior de Investigaciones Científicas (CSIC). Actualment a l'Institut hi fan recerca 21 Investigadors Principals (IP) i el personal dels seus grups, que sumen un total de més de 80 investigadors, entre pre-doctorats, post-doctorats, estudiants de màster, investigadors visitants i tècnics de suport a la recerca. Mentre es complerta un edifici específic, l'institut està provisionalment allotjat en dues seus: una al Centre Mediterrani d'Investigacions Marines i Ambientals (CMIMA) i l'altra al Parc de Recerca Biomèdica de Barcelona (PRBB).

Figura 1.1 - Logotip de l'Institut de Biologia Evolutiva.

El principal objectiu de l'Institut és la recerca biològica, representada pels seus 5 programes de recerca: Filogènia i Sistemàtica Animal, Evolució Funcional en Insectes, Sistemes Complexos, Genètica de Poblacions i Genòmica Comparativa i Computacional. Precisament aquests dos darrers camps s'han vist molt afectats per l'explosió que han viscut en els darrers anys les tecnologies de genotipació i seqüenciació. Aquesta expansió ha provocat un sobtat increment de les dades genètiques disponibles, tant públiques com privades, tot permetent la realització d'estudis científics utilitzant conjunts de dades molt més grans.

Degut a aquest increment continu del volum de dades necessàries, les capacitats de càlcul actuals de l'Institut s'han vist clarament sobrepassades, fent necessària una gran ampliació d'aquestes capacitats que permeti als investigadors seguir realitzant ciència de qualitat.

Objectius del projecte

Per tal d'ampliar la capacitat de càlcul de l'Institut, s'ha decidit instal·lar un sistema de càlcul d'alt rendiment orientat a aplicacions científiques que sigui capaç de satisfer les creixents demandes computacionals de tot el personal investigador de l'IBE. L'objectiu d'aquest projecte consisteix en realitzar el seguiment de totes les fases que suposen aquesta instal·lació, des de l'especificació i l'avaluació de requisits, fins a la implementació, avaluació i manteniment del sistema.

Per a poder entendre les necessitats i particularitats d'aquest sistema, caldrà, però, realitzar primer una breu introducció a la biologia en general, i a la genètica en particular, que ens permeti entrar en context.

(Breu) Introducció a la genètica

Tots els éssers vius estan composats per cèl·lules, i dins de cadascuna d'elles, al seu nucli, hi trobem els cromosomes, els quals contenen tota la informació genètica de l'individu. En el cas dels humans, el nostre genoma està format per 23 parelles de cromosomes (*Figura 1.2*). Cadascuna d'aquestes parelles està formada per 2 cromosomes pràcticament idèntics, un provinent del pare i l'altre de la mare, que són anomenats autosomes (els cromosomes numerats de l'1 al 22) i cromosomes sexuals (X i Y).

Cadascun dels cromosomes està format per una parella de "fibres" entrellaçades, la famosa "doble hèlix". Aquestes fibres, en estat de repòs, estan embolicades i empaquetades, però es desemboliquen mitjançant mecanismes moleculars per a poder obtenir la informació continguda en elles (*Figura 1.3*).

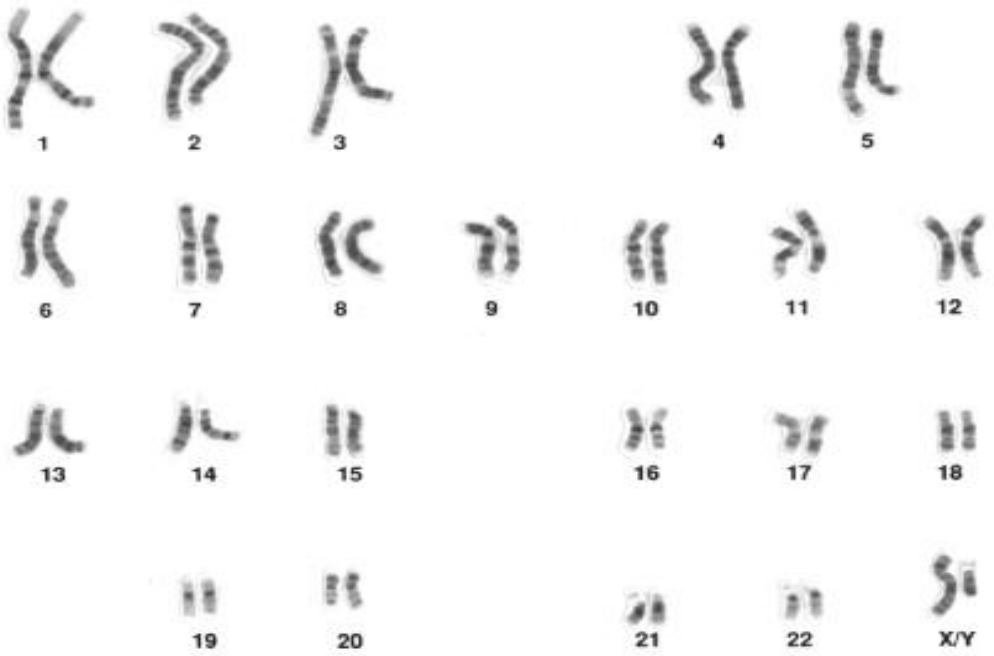

Figura 1.2 – Cariotip d'un home, mostrant-nos els 22 parells de cromosomes més els dos cromosomes sexuals. Font: Wikipedia.

Les unitats bàsiques que composen els cromosomes, i, per tant, tot l'ADN, són les bases hidrogenades, també conegudes com nucleòtids o, simplement, bases. Cadascuna d'aquestes bases és una molècula pertanyent a un conjunt de només 4 molècules (Adenina, Citosina, Guanina i Timina), les quals es representen amb els símbols A, C, G i T (*Figura 1.4*). Degut a que els humans tenim dues còpies de cada cromosoma, s'acostuma a tractar amb parells de bases enlloc de bases simples. És a dir, es treballa amb una parella de nucleòtids (AA, CC, GG, TT) que representa als dos nucleòtids que es troben a la mateixa posició a tots dos cromosomes, el procedent del pare i el de la mare.

El conjunt de tot el genoma humà està format per una seqüència d'aproximadament $3x10^9$ parells de bases distribuïdes entre els 23 cromosomes. De tota aquesta seqüència, només un 4% correspon als gens. El genoma humà conté aproximadament 23.000 gens clàssics, que són les seqüències encarregades de codificar proteïnes. La resta del genoma humà està format per seqüències reguladores, elements repetitius, elements mòbils i seqüències no codificadores.

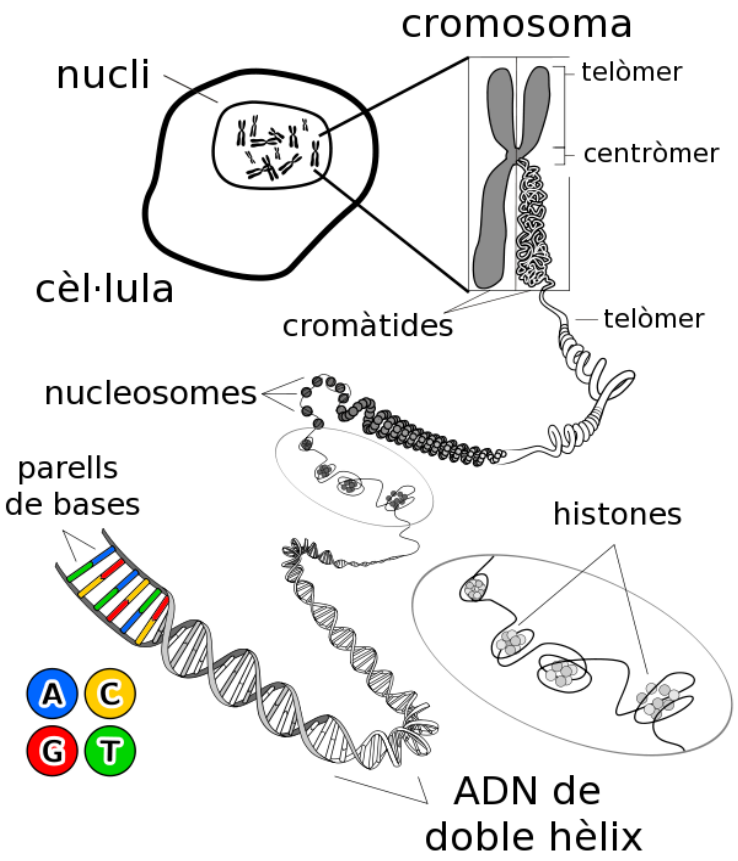

Figura 1.3 – Estructura d'un cromosoma. Font: Wikipedia (modificat).

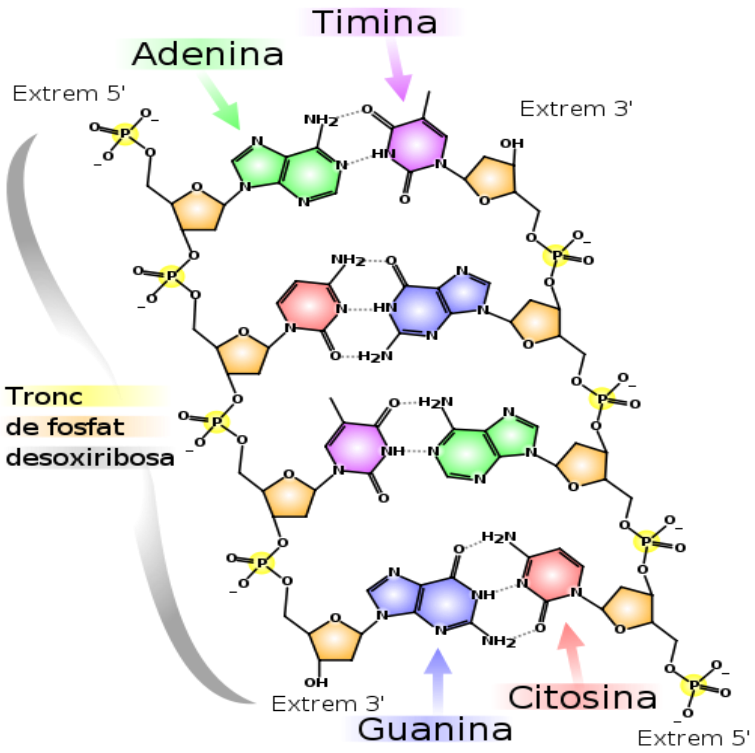

Figura 1.4 – Composició química i enllaç dels nucleòtids. Font: Viquipèdia.

Diversitat genètica humana

Tots els humans som diferents. Un individu pot ser de pell blanca, pel-roig, d'ulls verds, del grup sanguini 0 i tenir una certa predisposició genètica a patir un excés de colesterol. Un altre individu pot ser de pell negra, cabell negre, ulls marrons, del grup sanguini AB i ser intolerant a la lactosa. Cadascuna d'aquestes variabilitats s'anomena fenotip, i tot i tenir tantes diferències, tots dos individus són humans.

Com hem vist abans, el genoma humà està format per una seqüència de $3x10^9$ parells de bases. Tots els humans tenim aproximadament un 99.9% del genoma idèntic. El 0,1% restant conté les diferències que ens fan diferents. Les varietats presents en aquestes posicions s'anomenen al·lels, i són l'essència de la diversitat humana. Aquestes diferències tenen dos orígens: l'acumulació de mutacions, i la recombinació dels cromosomes a les cèl·lules germinals dels progenitors (les cèl·lules que donaran lloc als òvuls i espermatozous). Aquestes variacions s'anomenen polimorfismes quan algun dels al·lels es troba present en més d'un 1% de la població.

Dins del genoma, aquestes variacions consisteixen en la substitució d'un número variable de nucleòtids per uns d'altres, o la inserció o deleció d'un o més nucleòtids en una determinada posició (*Figura 1.5*). Per tant, els al·lels prenen diferents formes segons la seva estructura i longitud, i els principals són els *SNPs*, les *Segmental Duplications*, les transposicions, les delecions i els grans rearranjaments cromosòmics. Fins ara, però, la principal font d'estudi de la variabilitat humana està en els SNPs i en CNVs.

Els anomenats SNPs (*Single Nucleotide Polimorphism*) consisteixen en parells de bases que són diferents dins d'una parella de cromosomes. És a dir, els nostres dos cromosomes 18, que hem heretat, un del pare i un de la mare, són idèntics en un 99.9%. Això vol dir que un 99.9% dels parells de bases seran homozigots pel mateix al·lel que la resta de la població (idèntics: AA, CC, GG o TT), i un 0.1% seran o bé heterozigots (diferents: AT, CG, AG, ...), o bé homozigots però diferents del que la població sol tenir en aquella posició (TT on sol haver-hi AA). Així doncs, es considera que un nucleòtid és un SNP quan, per a aquesta posició genètica la freqüència de l'al·lel no-majoritari

supera l'1% en la població (és a dir, que un 1% o més dels humans tenen al·lels diferents).

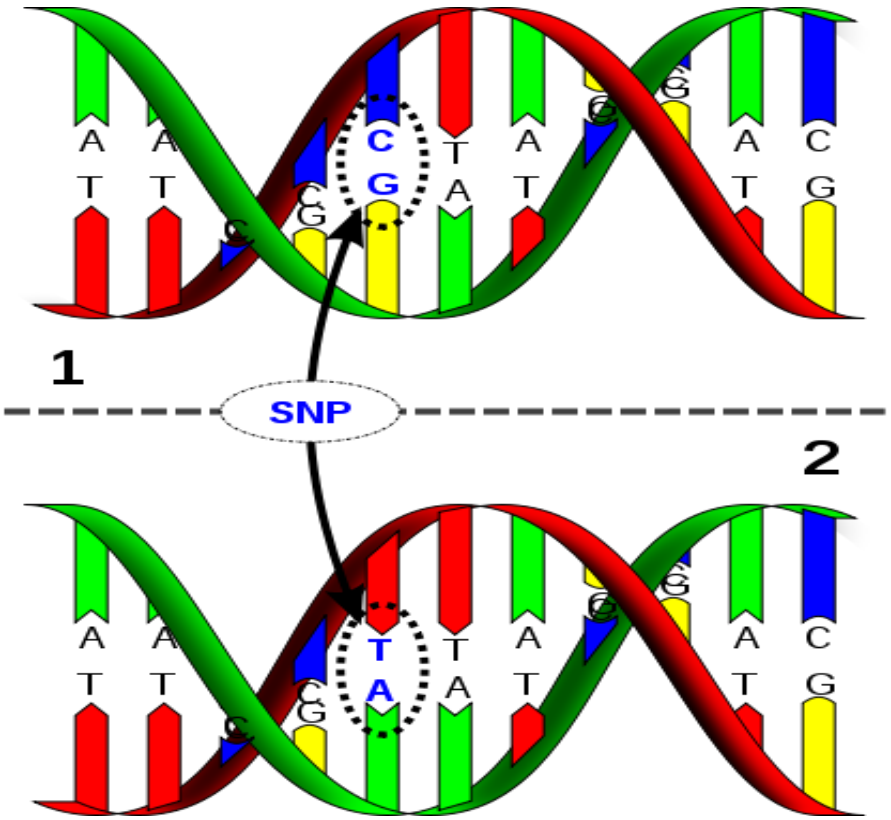

Figura 1.5 – Exemple de SNP on podem veure la substitució d'un dels parells de bases. Font: Wikipedia.

Per altra banda, hi ha segments de l'ADN que estan formats per diverses repeticions de cadenes més o menys llargues (entre un i diversos milions de parells de bases de longitud). Depenent de la longitud, aquestes repeticions reben noms diferents: microsatèl·lits (d'1 a 6 bases), minisatèl·lits (de 10 a 1.000) o *Copy Number Variations* (de 1.000 a diversos milions de bases). Aquestes duplicacions tenen diversos usos i motius d'estudi. Els microsatèl·lits i minisatèl·lits, per exemple, presenten una gran variabilitat i són utilitzats com a "empremta genètica" en genètica forense, però són responsables també de la malaltia de Huntington, la qual depèn del nombre de repeticions d'una seqüència de tres bases (CAG) al gen HD, les quals, si estan repetides consecutivament més de 35 cops, provoquen la malaltia. Pel que fa a les CNVs, tenen un gran interès d'estudi ja que es corresponen a regions molt grans, les quals poden incloure desenes de gens i altres elements (*Figura 1.6*).

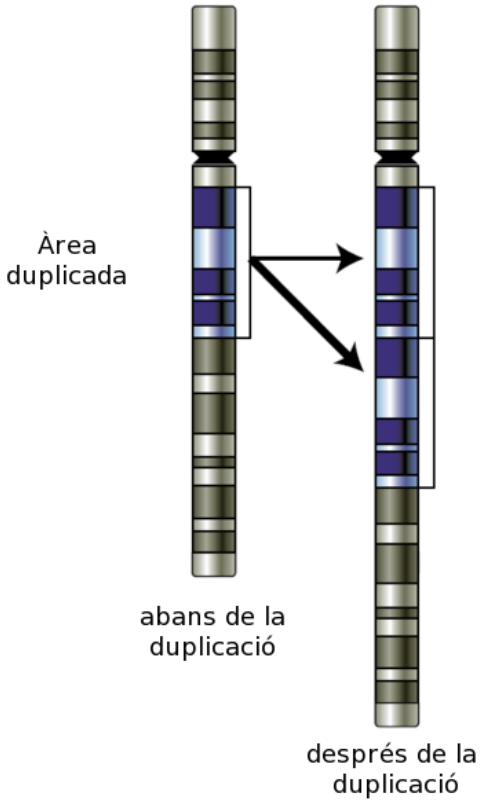

Figura 1.6 – Exemple de duplicació. En aquest cas es tracta d'una CNV (el segment duplicat és molt gran) que crea una segona còpia de tota una regió. Font: Wikipedia (modificat).

Genètica de poblacions

Tota aquesta variabilitat del genoma es transmet de pares a fills, canviant a cada generació, sotmesa a diverses forces evolutives. La genètica de poblacions és la branca de la biologia evolutiva que estudia la variabilitat que trobem entre diferents poblacions humanes. Aquesta variabilitat es veu afectada i determinada per la selecció natural (adaptació a les condicions d'un ambient determinat), la deriva (atzar), i per les forces de diferents efectes demogràfics, com són les migracions o l'efecte fundador (si 10 persones colonitzen una illa, la variabilitat genètica de tota la seva descendència dependrà de la d'aquests 10 individus en concret). L'estudi d'aquesta variabilitat mitjançant la genètica de poblacions permet conèixer la historia evolutiva de la humanitat (*Figura 1.7*).

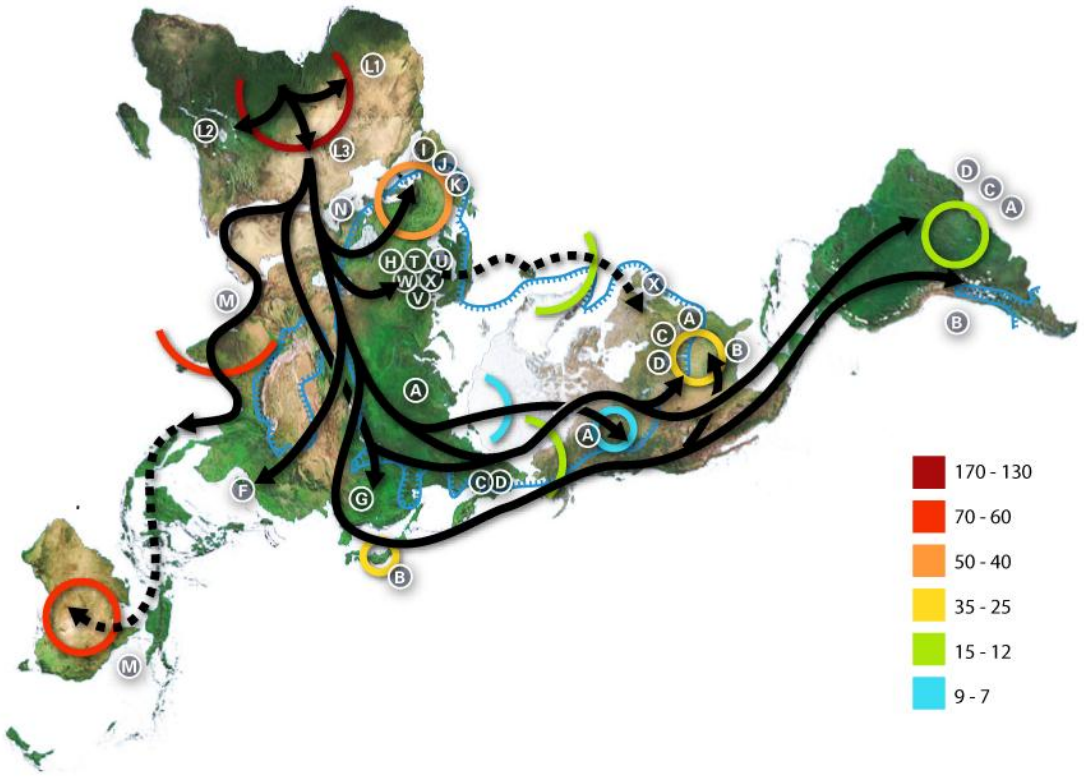

Figura 1.7 – Mapa de les migracions humanes basades en l'anàlisi de l'ADN. Els cercles de color indiquen milers d'anys fins al present. Font: Wikipedia.

Projectes

L'objectiu del sistema és que sigui disponible per a tots els projectes actuals i futurs de l'Institut que puguin necessitar del seu servei. Alguns dels projectes que esperem que es puguin beneficiar de la seva existència són, a tall d'exemple, els següents:

- **Cerca de marcadors genètics relacionats amb trets fenotípics en poblacions humanes:**
  Anàlisi de SNPs en diverses poblacions per tal de detectar variants associades a trets fenotípics.

- **The Evolution of Cooperarion and Trade (Eurocores):**
  Un projecte finançat per la ESF que pretén desentranyar les bases genètico-evolutives del comportament social humà.

- **La distribució mundial de freqüències al·lèliques en gens de malaltia:**
  L'estudi Malalties humanes i les seves implicacions evolutives: El rol de l'evolució recent en la replicació d'estudis d'associació.

- **Bases genètiques i víriques de l'Esclerosi Múltiple:**
  Estudi dels factors genètics i vírics que incrementen el risc d'Esclerosi múltiple, a més de la seva història evolutiva. El projecte implica també la col·laboració general en els projectes de la REMM (Red Española de Esclerosis Múltiple).

- **Detecció de selecció positiva dins de Variació Estructural (Duplicacions Segmentals i CNVs):**
  Estudi d'adaptacions entre membres de famílies de gens de recent aparició degut a la duplicació de segments.

- **CIBER en Epidemiologia i Salut Pública:**
  Col·laboració bioinformàtica general amb diversos grups del CIBEResp.

- **Factors de risc genètic en malària:**
  Aproximació bàsica a les xarxes en l'estudi de la susceptibilitat a la malària. L'estudi es basa en la Biologia de Sistemes, centrant-se en les xarxes de gens més que en gens individuals.

- **Dinàmica de la recombinació en el genoma humà:**
  Caracterització d'esdeveniments de recombinació en el cromosoma X en homes.

- **Caracterització genètica de la flora microbiana en pell sana i psoriàtica:**
  Reseqüenciació dirigida i metagenòmica bacteriana i viral en mostres de pell sana i psoriàtica.

Definició de requisits

Prefaci

Abans de començar l'anàlisi dels requisits del sistema, cal destacar un assumpte important. La planificació d'aquest sistema de càlcul va començar a l'any 2007, quan l'Institut de Biologia Evolutiva encara no era més que un miratge a l'horitzó. Degut a això, tot el projecte es va realitzar amb les necessitats, les dimensions i, sobre tot, el pressupost de la Unitat de Biologia Evolutiva de la Universitat Pompeu Fabra. Degut a això, les necessitats i avaluació que veurem en aquest apartat es van fer tenint presents unes dimensions més reduïdes del projecte, tot i tenir sempre en compte al futur Institut.

Propòsit del sistema

Durant els darrers anys, hi ha hagut una autèntica explosió pel que fa a les tecnologies de genotipació i seqüenciació, que han provocat un enorme increment en la quantitat de dades genètiques que es poden obtenir de cara a realitzar qualsevol tipus d'estudi. Addicionalment, degut a la popularització d'internet i a que la major part d'aquesta recerca s'ha realitzat utilitzant fons públics en universitats d'arreu del món, han sorgit diversos projectes com HapMap, 1000genomes, ensembl, dbsnp,… amb el propòsit de fer aquestes dades accessibles per a qualsevol que vulgui utilitzar-les.

Com a conseqüència d'aquests dos fets (l'increment del nombre de dades i la seva disponibilitat) han aparegut noves necessitats científiques (com més dades puguis fer servir en un estudi, els resultats obtinguts seran més significatius), i aquestes comporten noves necessitats computacionals de cara a poder estudiar i manipular aquestes dades. Degut a això, s'ha arribat a un punt on els usuaris no poden assumir l'execució de determinats càlculs en el seu ordinador personal, i es torna necessària la disponibilitat d'algun sistema de computació més potent on poder realitzar remotament aquests càlculs.

Addicionalment a aquestes necessitats, durant els darrers mesos al departament hi ha hagut diversos problemes relacionats amb la pèrdua de dades. Degut a això, hi ha un

interès en que aquest sistema, conjuntament amb la resta de tasques pel qual es vol fer servir, permeti als usuaris disposar d'un petit espai on realitzar-hi backup de les seves dades més importants.

Estat actual del sistema

Actualment, tots els anàlisis i càlculs que es realitzen al departament es realitzen mitjançant tres mètodes: utilitzant els ordinadors personals, llogant recursos a servidors externs (per exemple a *marenostrum*), o utilitzant *Snpator*, un servei del grup composat per un conjunt de servidors els quals posen a disposició dels usuaris dues eines web: Sysnps i Snpator.

Snpator[1] (http://www.snpator.com) consisteix en un servei web, inicialment desenvolupat per a donar suport als clients del Centro Nacional de Genotipado (CeGen), el qual permet pujar dades genotípiques en diferents formats, les quals s'emmagatzemaran en una base de dades MySQL, i que posteriorment permetrà descarregar-les en diferents formats preparats com a entrada formatada per a diversos programes d'ús comú. També permet executar aquests programes de manera remota mitjançant webservices, disposant d'un sistema de cues que controla l'execució de fins a 8 anàlisis simultanis.

L'estructura de hardware del sistema és la següent:

- Snpator03: servidor web. 2 processadors Intel Xeon 3.40GHz Dual Core de 64bits i 9GB de RAM.

- Snpator02: servidor de base de dades. 1 processador Intel Xeon 3.20GHz Dual Core de 64bits i 4GB de RAM.

[1] Morcillo-Suarez C, Alegre J, Sangros R, Gazave E, de Cid R, Milne R, Amigo J, Ferrer-Admetlla A, Moreno-Estrada A, Gardner M. et al. SNP analysis to results (SNPator): a web-based environment oriented to statistical genomics analyses upon SNP data. Bioinformatics (Oxford, England) 2008;24(14):1643–1644. doi: 10.1093/bioinformatics/btn241 http://bioinformatics.oxfordjournals.org/content/24/14/1643.long

- Un xassís IBM BladeCenter E amb 4 servidors blade: servidors de webservices. Cadascun d'ells format per 2 processadors Intel Xeon E5130 2.00GHz de 64bits i 2GB de RAM.

- Una cabina servidora de disc SAN (*Storage Area Network*) model IBM 4100 amb 8 discs de 250GB i 6 discs de 375GB servits per Fibre Channel, servint un total de 2,8TB de disc.

Aquest sistema, tot i acomplir perfectament la seva funcionalitat, ha demostrat no poder assumir el repte que proposen totes les noves dades genotípiques que ens arriben. S'han vist especialment compromesos el sistema de base de dades, el qual no pot manegar més de 4 milions de genotips simultàniament (ja que no es va pensar el sistema per a aquestes dimensions), i el sistema de webservices, ja que les màquines que els executen no poden tractar amb conjunts de dades tan grans.

Requisits del sistema

Partint del propòsit del sistema i del que tenim actualment, definirem les necessitats que haurà de cobrir el nou sistema.

Usuaris

Els usuaris potencials del sistema seran investigadors principals, post-docs i estudiants de doctorat en biologia. El principal ús que faran del sistema és l'execució de petits scripts en perl o python o programes fets per ells, i de programes creats per tercers. Les nostres previsions apunten inicialment a entre 10 i 20 usuaris del sistema de càlcul, i entre 40 i 50 pel sistema de backup, podent-se ampliar en un futur.

Software a executar

L'objectiu principal del sistema és permetre l'execució de software de tractament de dades genètiques i estadístiques. Aquests programes són d'ús freqüent en la recerca genètica i biologia evolutiva i la majoria d'ells han estat presentats en publicacions científiques. Alguns exemples d'aquests programes són:

- Blast
- Coalescent
- Phase
- Fastpahse
- Gap
- Haploview
- Ldhat
- Plink
- R
- Simcoal
- Staden

Aquesta és només una mostra d'alguns dels programes més coneguts que es solen utilitzar en els estudis al departament, però hi ha una gran diversitat de programes d'aquest tipus. Cadascun realitza una tasca concreta, que pot resultar interessant o necessària per a algun estudi en concret i que, degut a la natura de la recerca, no podem preveure en un moment inicial si es necessitaran o no, però hem de poder utilitzar-los en cas de fer falta.

Tot i aquesta diversitat de programes i funcionalitats, podem extreure algunes de les característiques que comparteixen:

- Desenvolupats en diversos llenguatges de programació: C, java, perl, python, …

- Versions per a tots els SO (ja sigui mitjançant binaris o distribuint el codi font). Això ens permet no limitar-nos a utilitzar un S.O. concret, però amb preferència per a sistemes GNU/Linux.

- Poc optimitzats, especialment en el tema de la gestió de memòria. En molts casos podria dir-se que estan programats sense pensar en la escalabilitat de les dades amb què tractaran.

- El temps transcorregut entre el disseny del programa i la seva publicació i popularització sol ser gran. Per tant, la majoria no permeten un ús acurat de la tecnologia actual. Per exemple, no tenen opcions per a fer ús dels processadors multi-core, tot i que en la majoria de casos suposaria una millora substancial. Tampoc solen ser ideats pensant en tractar amb grans volums de dades (hi poden treballar, però no es comptava amb ells com a requisits inicials). Hi ha excepcions en aquest apartat, especialment entre els programes més utilitzats i populars com són Blast i Phase que són mantinguts temps després de la seva publicació.

Addicionalment a tots aquests programes, també és necessària l'execució de scripts i programes realitzats en diversos llenguatges, principalment perl, python i java, creats pels usuaris per a poder realitzar anàlisis concrets i tractar amb les seves dades.

Una particularitat que cal destacar sobre les dades i, per tant, processos, que caldrà executar al sistema, és que sol tractar-se de dades multivariable. D'aquesta manera és molt comú que enlloc de ser necessari un gran anàlisi sobre el conjunt de totes les dades, es realitzin múltiples execucions d'un mateix procés sobre subconjunts de les dades. Per exemple, l'execució d'un mateix procés però dividint les dades segons la seva població, segons el cromosoma, o segons el fenotip que es busca estudiar, fent-les fàcilment paral·lelitzables per part de l'usuari.

Un tema en el que sí que volem aprofundir és en la capacitat de paral·lelització. Volem disposar de compatibilitat per als principals sistemes de paral·lelització disponibles. No pretenem que els nostres usuaris aprenguin a programar els seus scripts utilitzant threads, OpenMP o MPI, però sí que volem poder utilitzar les opcions multiprocessador dels programes que estiguin preparats per a usar-les.

Tipus de dades

El tipus de dades que es fan i es faran servir els podem classificar en 3 categories: Arxius de text, arxius binaris i bases de dades.

Arxius de text

Fins ara, els estudis que s'han estat realitzant generalment utilitzen dades sobre SNPs relativament petites, amb magnituds d'entre centenars a un miler d'individus per uns pocs centenars de SNPs. Tenint en compte que les dades sobre SNPs es representen amb dues lletres, això ens donava per norma general uns arxius de text tabulat d'entre 2 i 5 MB. Però ara, degut als avenços de les tecnologies de genotipació (per increment de la capacitat i velocitat de genotipat) i l'abaratiment de costos, aquestes magnituds han canviat molt, i això té un impacte directe en la dimensió de les dades. Alguns d'aquests avenços són:

- **GWAS:**

Els GWAS (*Genome-Wide Association Studies*) consisteixen en uns nous xips de genotipat que permeten obtenir fins a 1.000.000 de SNPs per individu. Si tenim en compte que el nombre de mostres que es fan servir en els estudis també va creixent, ens trobem amb arxius d'entre 3 i 7 GB. L'anàlisi d'aquests volums de dades, a part de ser cada cop més costós computacionalment, comporta problemes de memòria (alguns dels programes que es necessiten utilitzar ho carreguen tot a memòria, sense preocupar-se de fer-ne cap gestió), i també problemes amb programes compilats per a arquitectures de 32bits (límit de 4GB d'adreçament de memòria).

- **Ultra-seqüenciació:**

Fins fa relativament poc, seqüenciar genomes complets resultava extremadament car i costós. Amb els avenços tecnològics que hi ha hagut al camp de la seqüenciació, però, cada vegada ho és menys. Degut a això estan sorgint arreu del món diversos projectes com per exemple el "*1000 genomes project*" que pretén posar a disposició pública 1000 genomes humans complets. Cadascun d'aquests genomes ocupa aproximadament 3 GB. Aquests genomes complets són molt útils en la execució d'estudis d'alineament de seqüències fent servir programes com, per exemple, Blast.

Arxius binaris

L'ús d'arxius binaris actualment no és gaire comú en la realització d'anàlisis al departament. Aquests, però, sí que són força usats en la realització d'estudis sobre CNVs, per exemple, on no ens serveixen les cadenes de text del genoma, sinó que cal fer servir les gràfiques d'intensitat sense processar que proporcionen les màquines de genotipat. Aquestes estan emmagatzemades en binari en un format propietari depenent del fabricant de la màquina i tenen un volum d'aproximadament 1.5TB per cada 1.000 individus.

Bases de dades

Aquest conjunt inclou una gran diversitat de dades. Des de bases de dades SQL pròpies de cada usuari i de seqüències de Blast, fins a bases de dades on-line públiques com HapMap, dbSNP o ensembl.

Serveis

Addicionalment a l'execució de processos, necessitarem que el nou sistema ofereixi un seguit de serveis, tant destinats al càlcul com al departament en general:

- **Web:**
  Servidor web que permetrà als usuaris disposar d'un website acadèmic personal per a publicar els seus resultats.

- **Base de dades del sistema:**
  Servidor de base de dades per a ser usat des dels nodes de càlcul.

- **Base de dades web:**
  Servidor de base de dades destinat a ser usat des del servidor web.

- **Samba:**
  Servidor Samba per a permetre als usuaris realitzar còpies de seguretat, així com facilitar l'accés a les seves dades des dels seus ordinadors personals.

Degut a la importància d'aquests serveis, creiem convenient que siguin servits des d'un sistema d'alta disponibilitat, el qual ens permetrà disposar d'aquest serveis en tot moment.

Control del sistema

Degut a que esperem que aquest sistema sigui capaç d'executar una gran quantitat de processos i serà utilitzat per diversos usuaris simultàniament, necessitarem que el sistema disposi de mecanismes de control per tal de gestionar els recursos i procurar-ne un correcte ús per part dels usuaris.

Requisits de hardware

Volem un sistema que permeti executar remotament en un servidor tots els càlculs que no puguin assumir els usuaris al seu PC. Això inclou aquells processos que consumeixen molts recursos i durant molt temps, però també aquells que demanen una quantitat de recursos que un ordinador personal no pot oferir (sigui memòria, disc o temps de càlcul), així com també processos més "petits", però que cal executar múltiples vegades.

Partint d'això, podem extreure algunes de les necessitats principals del nostre sistema:

- **Nombre de processadors**

  Degut al nombre d'usuaris i processos que podran estar utilitzant el sistema concurrentment, cal un alt nombre de processadors.

- **Potència dels processadors**

  Quanta més, millor, però no és una prioritat. El que sí que és important és que siguin tots d'arquitectura x86_64 bits, per a poder utilitzar els binaris i compilar per a aquesta arquitectura i evitar així els problemes de limitació de memòria (4GB) i de mida màxima d'arxius que poden sorgir.

- **Memòria**

  Hem de ser capaços d'executar diversos processos alhora. Per tant, hem de poder proveir de memòria a tots ells. Alguns programes consumeixen molta memòria, i d'altres pràcticament gens, però hem de poder satisfer les necessitats de tots dos tipus. Demanarem un mínim d'1GB memòria per procés/core.

- **Capacitat d'emmagatzematge**

  Les necessitats de disc per a determinats projectes poden ser realment grans, i podem estar treballant amb diversos projectes a la vegada. Cal tenir una capacitat d'emmagatzematge que sigui capaç de satisfer-les.

- **Connectivitat dels usuaris**

  Els usuaris del sistema han de poder accedir-hi en qualsevol moment i des de qualsevol lloc per a executar-hi els seus processos, o recollir-ne les dades. D'aquí extraiem el següent requisit.

- **Tolerable a fallades**

  El sistema ha de ser fiable. Tot sistema és tan vulnerable com el seu element més dèbil, i en un sistema complex, el nombre d'elements és molt gran, Cal assegurar que els problemes amb un determinat element afectin al sistema el mínim possible.

- **Escalable**

  El departament té una gran necessitat d'aquest sistema, i preveiem que serà de gran utilitat a mig i llarg termini. Així, en funció del seu rendiment i de les necessitats del moment, el sistema s'anirà ampliant a mesura que arribin nous projectes i inversions. Per a això, necessitem un sistema escalable que pugui anar creixent progressivament, a la vegada que aprofitem tot el que ja s'ha instal·lat.

Limitacions

A l'hora de dissenyar el sistema, hem de tenir en compte també les limitacions que tenim per compatibilitat amb el hardware de que disposem actualment al departament, així com altres limitacions físiques que condicionaran les seves característiques i ubicació.

Hardware

Disc:

Disposem al departament d'una cabina de disc IBM model DS4100, amb connexió Fibre Channel i doble controladora, amb els 14 slots de disc plens repartits en 8 discs de 250 GB i 6 discs de 375 GB. Serveix un total de 2375GB (un cop descomptat l'espai requerit pels discs de hot-spare i replicació en RAID) distribuïts entre 9 particions muntades als servidors que disposem al departament actualment. Aquesta cabina inclou el seu propi software de control "IBM System Storage DS4000/FAStT Storage Manager 10", el qual permet controlar l'estat de la cabina així com manegar els volums lògics, assignació d'interfícies, etc.

Connectivitat

Xarxa:

La velocitat de la connexió de xarxa que fem servir per a interconnectar les màquines està limitada per la velocitat dels switchs ethernet 1 Gbps dels que disposa actualment la xarxa de la Universitat. Aquests switchs són els encarregats d'intercomunicar totes les xarxes internes amb la resta de xarxes de l'edifici, universitat i exterior. Fem el que fem, la nostra estructura de xarxa haurà de passar per algun d'aquests switchs.

Fibra:

La cabina de discs de que disposem al departament té connexió de fibra òptica Fibre Channel a 1Gb. Ara mateix totes les connexions ja estan ocupades per la resta de servidors del departament. Per tant, tant si volem utilitzar aquesta cabina des del nou sistema com si volem utilitzar noves connexions de fibra des dels servidors actuals, caldrà disposar d'un nou switch de fibra òptica per a poder realitzar les connexions.

KVM:

Disposem d'un terminal KVM (*Keyboard-Video-Mouse*) per a accedir als servidors que tenim actualment al departament. El nou sistema haurà de ser compatible amb ell per tal de facilitar-ne la gestió.

Físiques

La sala on s'instal·larà el sistema és la sala de servidors del Parc de Recerca Biomèdica de Barcelona. A l'hora de dissenyar el nostre sistema, cal que tinguem en compte les característiques i limitacions d'aquesta:

Espai:

En aquesta sala disposem de 4 *racks* (armaris) per al nostre grup, un d'ells està especialment habilitat per a servidors i xassís de blades, en el qual hi disposem de 15U lliures (*U* és la unitat en què es mesura l'alçada dels components que s'instal·laran en un rack. 1U equival a 1,75 polzades, o 44,45mm. 15U suposen 66cm lliures). Els altres 3 estan força desocupats, i també tenim la possibilitat d'instal·lar-hi un cinquè rack.

Instal·lació elèctrica:

La instal·lació elèctrica de la sala de servidors de l'edifici alimenta cada rack amb dos circuits que provenen de dos SAIs diferents (trifàsics de 200 KVA i 600 KVA). Els quadres elèctrics poden suportar potències de 250 A i 280 A, a 220 V. Aquests SAIs estan recolzats per grups electrògens en cas de tall del subministrament elèctric per part de la companyia.

Gràcies a tot això, disposem d'una gran robustesa en cas de fallada elèctrica, la qual considerem imprescindible. Per això ens imposem com a requisit per a tots els elements del sistema la presència de fonts d'alimentació redundants que incloguin balancejadors de càrrega per a distribuir-ne l'ús.

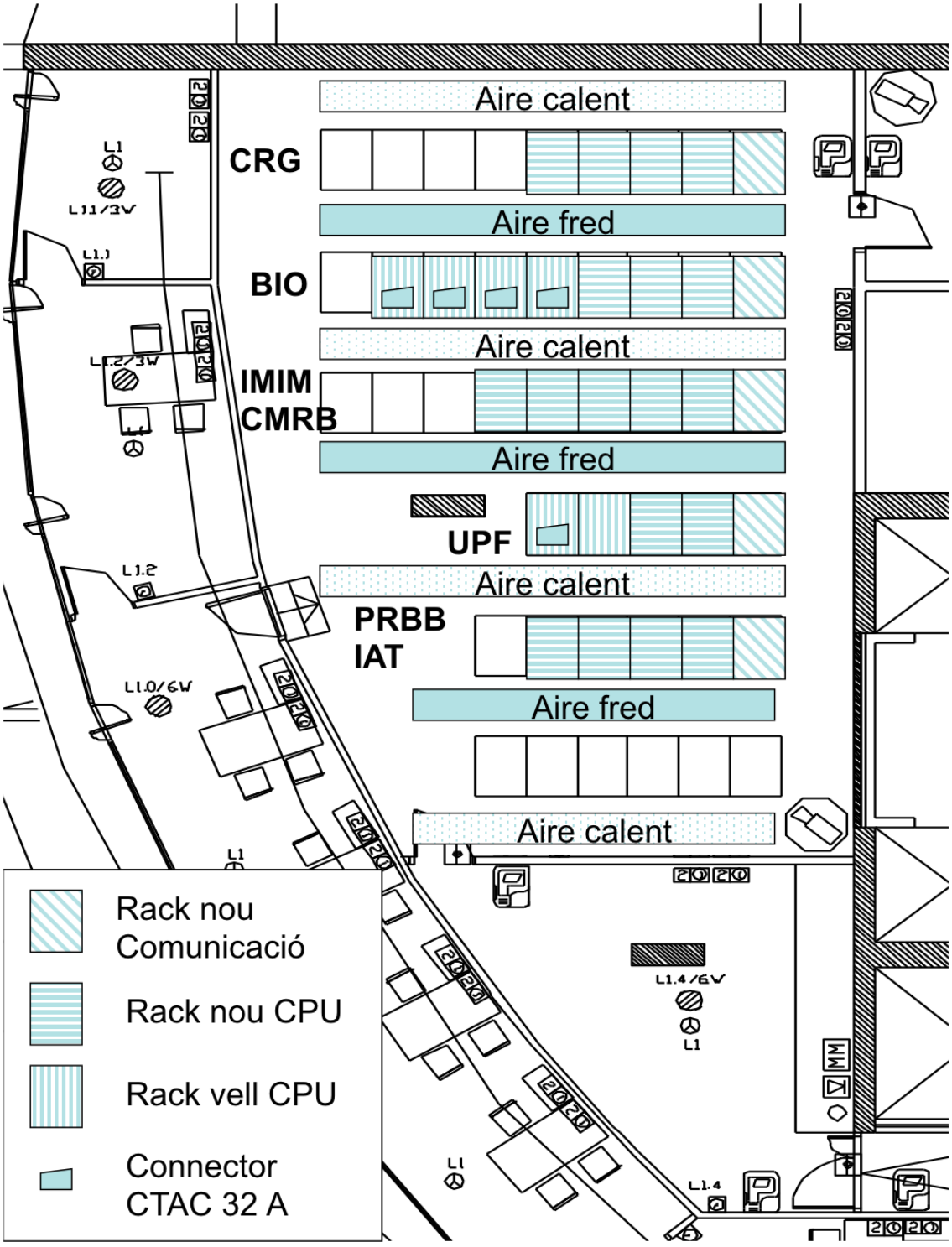

Figura 2.1 - Plànol de la sala de servidors del PRBB. Font: PRBB.

Consum elèctric:

No tenim cap limitació en quant al consum elèctric d'aquestes màquines, per lo qual no cal que ens imposem restricció al respecte, però intentarem que el sistema faci un ús raonable de l'energia per a reduir-ne la seva petjada ecològica.

Potència de refrigeració:

La sala on es situarà el nostre sistema està refrigerada per quatre equips: dos per aigua freda de 35 KW (30,25 frigories/hora) cadascun, i dos d'expansió directa de 52 KW (45 frigories/hora) cadascun.

Pressupost

El pressupost del que disposem per a la compra de nou maquinari és d'aproximadament 95.000€. Aquesta compra s'haurà de realitzar seguint els protocols de la universitat.

Alternatives al mercat

A continuació analitzarem les diferents alternatives que hi ha al mercat, per tal d'escollir el sistema que millor s'adeqüi a les nostres necessitats.

Estructura

Existeixen al mercat diverses estructures de hardware que ens permeten cobrir les necessitats que tenim. Tot i aquesta diversitat, no hi ha un gran consens respecte a la nomenclatura de les diferents estructures, i les fronteres entre elles poden resultar difuses amb estructures intermitges o híbrides. Aquestes són les principals:

Supercomputador:
Hardware especialitzat en què múltiples components estan connectats a un únic bus de sistema controlat per un únic sistema operatiu. El hardware que es fa servir és altament especialitzat, i en moltes ocasions s'utilitzen sistemes operatius especials. Per exemple, la sèrie Power 7 de IBM inclou sistemes de fins a 256 processadors i 8TB de memòria RAM en una única màquina. Aquests sistemes són molt cars i estan especialment pensats per a processos altament paral·lels que necessiten de la velocitat del bus únic.

High-Performance Cluster (HPC):
Un clúster és un conjunt de màquines independents, interconnectades de manera que poden treballar juntes. Les màquines poden ser idèntiques (clúster homogeni) o diferents (heterogeni), i poden utilitzar arquitectures d'ordinador tradicional, o hardware més específic (múltiples processadors, gran quantitat de ranures de memòria, etc.).

Els clústers d'alt rendiment solen ser homogenis, utilitzen hardware especialitzat i el seu objectiu principal és realitzar l'execució de processos paral·lels el més ràpidament possible. Per això, utilitzen sistemes d'interconnexió altament especialitzats entre els diferents nodes, com són myrinet o infiniband. Resulten més lents que els supercomputadors "monolítics", però molt més escalables, ampliables i el seu rendiment pot ser millor segons el tipus de procés.

High-Throughput Cluster:

Els clústers *high-throughput* consisteixen, a l'igual que els d'alt rendiment, en un conjunt de màquines interconnectades. El seu objectiu, però, enlloc de consistir en l'execució ràpida d'un únic procés paral·lel, consisteix en l'execució continuada de múltiples processos més simples. Per això, permet utilitzar sistemes d'interconnexió menys eficients (però més assequibles) com ethernet.

High-Availability Cluster:

Els clústers d'alta disponibilitat consisteixen en un conjunt reduït de nodes, interconnectats mitjançant una capa de software la qual els permet replicar un conjunt de serveis entre tots els nodes del clúster. És a dir, en cas de fallada d'algun node, sempre n'hi haurà algun altre capaç de mantenir tots els serveis actius.

Grid:

L'estructura d'un grid és similar a la d'un clúster, però sol estar orientada a l'espectre més baix i heterogeni de la gamma. El seu objectiu és l'execució de processos individuals no excessivament complexos, de manera que requereixen d'una comunicació molt més relaxada, fins al punt de poder-se constituir amb màquines connectades via internet, i que només treballen en els moments de poca càrrega. Un exemple famós d'aquests és el projecte SETI@home, el qual analitza dades de radiotelescopis cercant senyals de vida intel·ligent procedent de l'espai, mentre els ordinadors dels milions d'usuaris estan inactius.

Coneixent totes aquestes estructures, decidim que la que millor s'adapta a les nostres necessitats és la d'un clúster, concretament de tipus *High-Throughput*, ja que el principal ús que farem d'ell és l'execució de batchs de jobs independents. Si bé és possible que també necessitem executar processos paral·lels, aquests no seran la nostra principal prioritat. D'aquesta manera, el disseny escollit ens permetrà executar-los igualment, però l'estalvi que suposarà el pas de xarxes especialitzades a una xarxa ethernet, ens permetrà realitzar una major inversió en CPUs, memòria i disc.

Addicionalment a aquesta estructura pel clúster de càlcul, creiem necessària també la creació d'un clúster d'alta disponibilitat de dos nodes, el qual serà l'encarregat de mantenir tots aquells serveis comuns i fonamentals per al funcionament del sistema.

Servidors

Pel que fa als servidors amb què construirem aquest clúster, i gràcies a la infraestructura de que disposem a la sala de servidors del PRBB, decidim utilitzar hardware homogeni i específic per a clústers. D'aquesta manera obtindrem una gran densitat de hardware respecte a l'espai, i també podem utilitzar els avantatges com les fonts d'alimentació redundants.

Dintre d'aquest mercat hi tenim dues alternatives principals: els servidors de tipus "*blade*", i els de tipus "*pizza*". Tots dos tipus consisteixen en màquines altament integrades i compactes. La principal diferència entre ells consisteix en la seva connectivitat: Mentre que els servidors de tipus "pizza" consisteixen en màquines individuals amb totes les seves connexions per a ser usades lliurement, els servidors de tipus "blade" necessiten d'un xassís on connectar-se (*figura 3.1*). Aquest xassís inclou un bus de connexió entre blades i s'encarrega de centralitzar totes les connexions, incloent els switchs ethernet i de fibra i les fonts d'alimentació de tot el sistema, entre d'altres. L'inconvenient principal del sistema de blades és que requereix de la inversió inicial d'un xassís, i la seva escalabilitat es veu més limitada que la de les "pizzes" (si omplim un xassís, per a poder instal·lar un nou servidor caldrà comprar un segon xassís), però a canvi ens dóna un sistema molt més senzill d'administrar.

Aquest tipus de servidors permeten instal·lar múltiples CPUs (entre 2 i 4) i gran nombre de plaques de memòria, de manera que, conjuntament amb els processadors de quatre nuclis, ens donaran una densitat de nodes molt elevada, oferint-nos la possibilitat d'executar un major nombre de processos en un menor nombre de màquines.

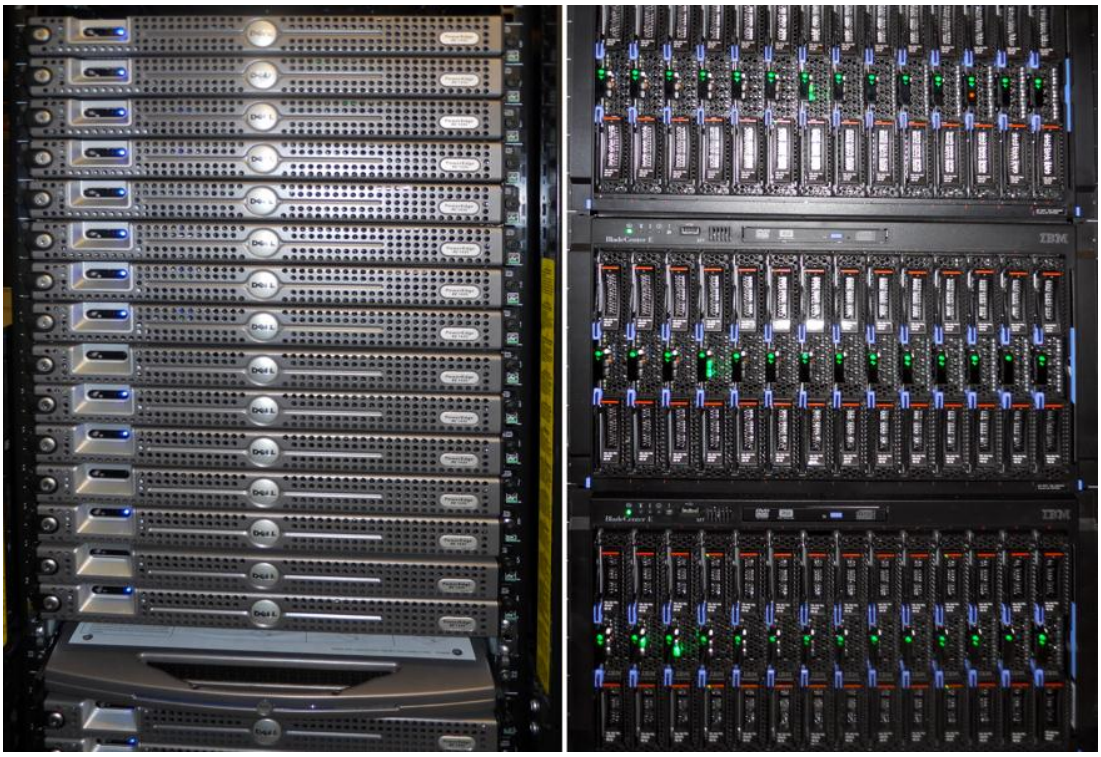

Figura 3.1 – Frontal dels dos tipus de servidors, pizza (esquerra) i blades (dreta). Podem veure la major densitat de nodes del tipus blade.

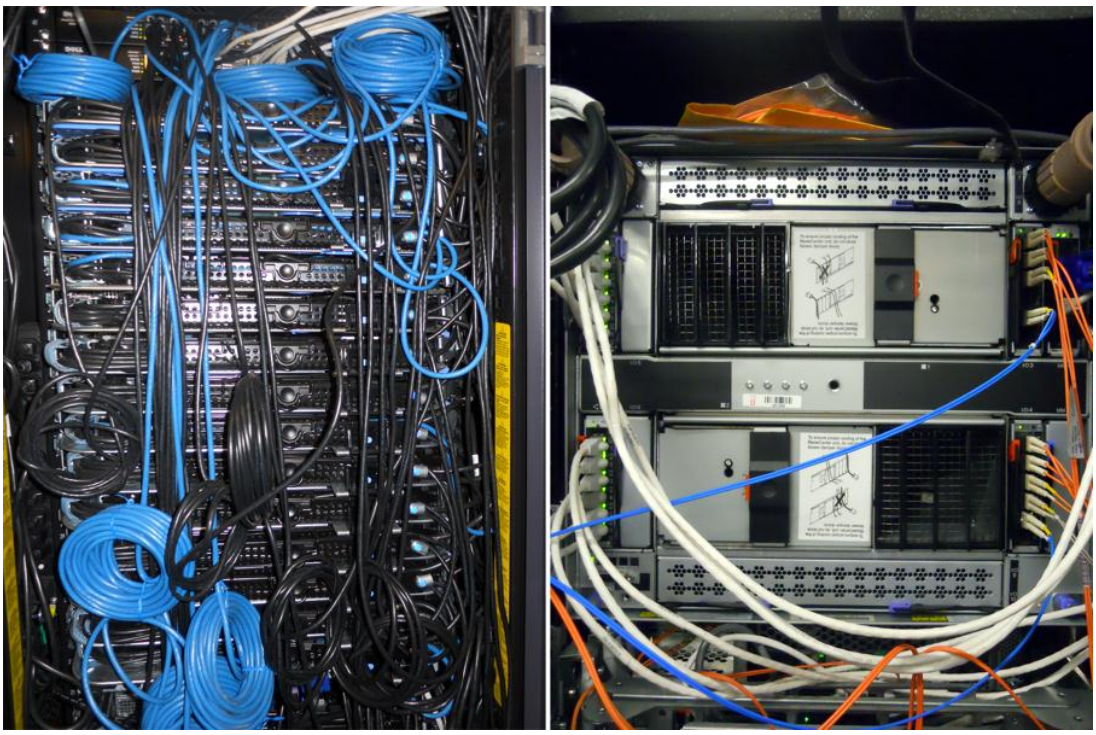

Figura 3.2 – Part posterior dels servidors de tipus pizza (esquerra) i blades (dreta). Les connexions que surten dels blades pertanyen ja als switchs de xarxa i fibra integrats.

Disc

Pel que fa a l'emmagatzemament de disc del sistema, els nostre objectiu és doble: necessitem molt espai de disc, i aquest espai ha de ser visible per tots els nodes. Per tal d'assolir això, hem d'actuar per dues vies, els discs físics i el sistema de fitxers.

Hardware

Dintre de les opcions de hardware, se'ns presenten diverses alternatives per tal d'escollir el tipus de suport físic que millor s'adapti al que necessitem, especialment pel que fa al volum dels discs.

Discs locals

La manera més simple de disposar d'un gran espai de disc és dotar a cada node del clúster d'un gran disc intern. Això però, presenta diversos problemes, i és que aquest disc, inicialment, només serà visible des del node i haurà de ser compartit tant pel sistema operatiu com per les dades. Aquests discs integrables al blade són molt cars respecte a l'espai que ofereixen, i en cas de necessitar ampliar l'espai de disc, no podem simplement afegir un disc nou, sinó que caldria substituir algun dels discs per un de més gran, o comprar un nou blade amb el seu corresponent disc. Però, addicionalment a tots aquests problemes, el principal inconvenient que presenta aquesta alternativa és la manca de redundància per hardware, per tal de protegir les dades.

SAN (*Storage Area Network*)

Els sistemes SAN, també coneguts com "cabines de disc", consisteixen en sistemes servidors de disc dedicats i amb espai per a múltiples discs, els quals serveixen diversos volums de disc a través de diversos tipus de connexions, com per exemple fibra òptica. Aquests sistemes s'encarreguen de tota la gestió dels volums i inclouen de sèrie diverses mesures de seguretat i redundància com controladores dobles, redundància per RAID o discs de Hot-Spare (discs de substitució ràpida i automàtica en cas de fallada mecànica d'un dels discs). Al departament ja disposem d'una cabina d'aquest tipus IBM 4100, però se'ns ha quedat petita i necessitem molta més capacitat de disc. El principal inconvenient que presenten aquest tipus d'elements és el seu elevat preu per gigabyte útil, ja que els discs són cars i es perd molt espai per redundància, però ho compensa la seva seguretat. A més, el sistema és força escalable ja que cada cabina inclou slots per a

múltiples discs que es poden anar afegint progressivament quan sigui necessària una ampliació d'espai.

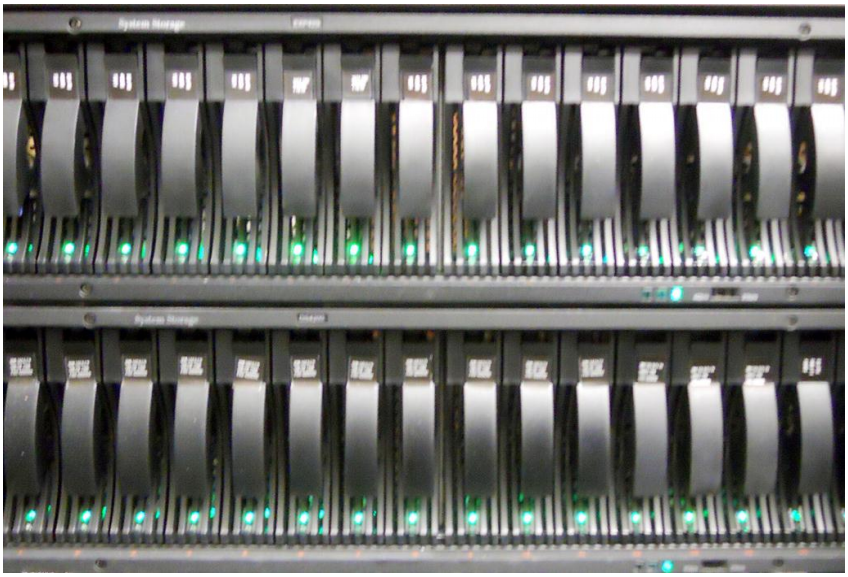

Figura 3.3 – Exemple de cabina de disc SAN.

NAS (*Network Attached Storage*)

Els sistemes NAS són una variant del les cabines de disc SAN, les quals, a part de totes les característiques i capacitats d'una cabina de disc tradicional, inclouen un servidor de sistemes de fitxers distribuïts utilitzant protocols com NFS o CIFS. Aquesta opció resulta molt interessant, ja que cobreix perfectament les nostres dues necessitats, tot i que, malauradament, al moment de la creació del clúster suposaven una tecnologia massa cara com per a poder permetre'ns-la.

Sistema de fitxers

Per tal de fer accessible els volums de discs a tots els nodes, caldrà usar un sistema de fitxers que ens ho permeti. Actualment hi ha diversos sistemes de fitxers que permeten l'accés a un mateix volum de disc des de múltiples nodes. Per tal de dur a terme això, hi ha dues grans famílies de sistemes de fitxers distribuïts segons el seu nivell d'accés:

Nivell de fitxer:

Sistemes formats per un servidor el qual munta físicament el volum de disc, i després el serveix a tots els clients, gestionant els accessos i col·lisions a nivell de fitxer. Impliquen un major trànsit de xarxa per a les transmissions, però la configuració i

topologia és més simple. Exemples: NFS, Samba/CIFS (per a sistemes Linux+Windows).

Nivell de bloc:

Sistemes formats generalment per un servidor de metadates i un o diversos servidors de disc. Els clients accedeixen al sistema de fitxers bloc a bloc, com en un sistema de fitxers tradicional. Exemples: OCFS, GPFS, Lustre.

Teòricament els sistemes a nivell de bloc tenen un millor rendiment que els de nivell de fitxer, però aquests resultats poden variar molt entre una configuració i una altra i segons el tipus d'aplicació. A més, els sistemes com NFS fan servir cachés tant al client com al servidor, de manera que el seu rendiment pot acabar resultant superior als altres. De cara al nostre sistema, optarem per a utilitzar el clúster d'alta disponibilitat (el qual estarà connectat per fibra òptica a la cabina SAN) com a servidor de fitxers NFS per a tots els nodes del clúster de càlcul.

Distribució de l'espai de disc

Les necessitats de disc que hem de suplir són:

- **Espai de sistema:**

L'espai de disc necessari on instal·lar el sistema operatiu, llibreries, fitxers de configuració i logs de cada servidor. Individual per a cada màquina i en el seu disc local.

- **Homes:**

Espai que contindrà els homes dels usuaris i on hi emmagatzemaran les seves dades. Ha de ser el més gran possible, i ha de ser accessible des de tots els servidors. Com és necessari fer backup d'aquestes dades, però no és possible assumir un backup de totes elles (per volum i temps), decidim partir aquest espai en dos volums: un volum "homes" petit, limitat per quotes i del que es farà backup, i un volum "scratch" molt gran i sense quotes, tot i que controlat (per a evitar que un sol usuari ompli el disc), del qual no es farà backup.

- **Webs personals:**

Espai on els usuaris tindran els fitxers de la seva web acadèmica personal. Estarà dins del home de l'usuari.

- **Espai comú:**

Espai on s'emmagatzemaran dades comunes per a tots als usuaris (per exemple, programes o llibreries) i que haurà de ser accessible des de tots els servidors.

- **Espai de base de dades:**

Espai on emmagatzemar les dades del servei de base de dades, tant del servidor de base de dades de "càlcul" com de les bases de dades "web".

- **Disc local dels nodes de càlcul:**

El disc local té velocitats d'accés molt més ràpides que les particions NFS, però, per contra, molt més petits que s'han de repartir entre el sistema operatiu, la swap i l'espai local, de manera que només es podrà utilitzar aquest espai per a processos concrets que requereixin d'una entrada/sortida molt ràpida, però amb poc volum de dades.

- **Backup dels PCs dels usuaris:**

Espai on els usuaris podran emmagatzemar els backups dels seus ordinadors de sobretaula, mitjançant particions compartides via samba. Ens interessa tenir-ho separat de l'espai de homes, ja que al fer backup dels homes no volem haver de fer backup dels seus backups, ja que seria redundant i una pèrdua d'espai i temps. Estarà limitat per quotes per evitar que algun usuari abusi de l'espai de manera incontrolada.

- **Backup dels homes**

Espai on realitzar el backup de les dades que els usuaris tenen als seus homes. D'això s'encarregarà el robot de cintes que té el departament.

Software

Els únics requisits que tenim de cara al sistema operatiu i software que correrà al sistema són: que sigui capaç d'executar tot el que ens faci falta, que accepti les diferents llibreries i mecanismes de programació paral·lela i que disposi d'un sistema gestor de recursos. D'aquests tres requisits, els dos primers tenen fàcil solució utilitzant gairebé qualsevol distribució de GNU/Linux, motiu pel qual ens centrarem en el tercer requisit, la gestió de recursos.

Middleware

El *middleware* (o sistema gestor de recursos, *Distributed Resources Manager System*, *Batch System*, *Job Scheduler*, sistema de cues, ...) consisteix en una capa de software que es situa entre el sistema operatiu i les aplicacions, la qual s'encarrega de fer una abstracció de tots els nodes i recursos que composen el sistema, distribuint i gestionant l'execució de tots els processos del sistema. Hi ha diversos tipus d'aproximació a aquest tipus de sistemes, des d'aquells que virtualitzen tots els recursos pertanyents al sistema creant la impressió de tenir un únic servidor (i encarregant-se el sistema de distribuir la càrrega i processos), fins aquells que respecten la independència de cada node i, simplement, s'encarreguen de gestionar-ne els recursos de manera remota. Aquest darrer tipus és el que més ens interessa, i en tenim diverses alternatives al mercat:

- PBS Works Suite:
  Desenvolupat per la NASA a principi dels 90 i posteriorment comprat per l'empresa Altair Engineering. És software propietari. Cost de les llicències: 14.25 € / any / core.

- OpenPBS:
  Versió Open Source del PBS original. La darrera versió és de juny del 2001.

- Torque:
  Open Source. "Hereu" d'OpenPBS.

- Sun Grid Engine:

  Open Source. Té un enorme recolzament, està molt estès, s'està millorant constantment i hi ha un gran suport tècnic a les llistes de correu.

- Platform LSF:

  Software propietari. 3 tipus de llicencia (per cpu):

  - Classe-B (fins a 2 CPUs i 4 GB de memòria)
  - Classe-S (fins a 4 CPUs i 16 GB de memòria)
  - Classe-E (enterprise, sense restriccions)

  Degut a les necessitats de memòria del sistema que instal·larem, ens veuríem obligats a fer servir llicències de Classe-S, i, en un futur, canviar-les possiblement per Classe-E, les quals són molt cares.

- Condor:

  Open Source. Permet la gestió simultània de recursos tant de servidors dedicats al càlcul, com d'ordinadors d'escriptori quan no s'estan fent servir formant xarxes tipus *grid*.

Veient que tots els sistemes cobreixen perfectament les nostres necessitats bàsiques i tots ells resolen sobradament la distribució de recursos del sistema, i que les principals millores d'uns respecte als altres són add-ons i opcions avançades que no sabem si necessitarem o no, ens decantem per una de les opcions Open Source. Concretament optarem per SGE[2], ja que veiem que és la que té més suport i alguns membres d'altres departaments de la Universitat treballen amb ella i ens poden aconsellar en cas de tenir algun problema.

Per altra banda, en cas de que el SGE no compleixi les nostres expectatives o necessitats, decidiríem canviar-nos a algun altre sistema, però ja amb més experiència i sabent exactament què necessitem.

[2] Aquesta decisió va ser presa molt abans de la compra de Sun per part d'Oracle, i la posterior "privatització" de SGE.

Sistema Operatiu

SGE permet treballar sobre diversos sistemes operatius, Linux inclòs. D'entre totes les distribucions de Linux, podem escollir diverses de les més famoses i a les que hi estem més acostumats, com RedHat, Debian, SuSE, OpenSuse, Ubuntu, ... però hi ha una distribució que ens ha cridat molt l'atenció: *Rocks-cluster*.

ROCKS és una distribució de Linux basada en CentOS, i que està totalment enfocada al seu ús en clústers. Inclou diversos paquets de sèrie, com diversos sistemes gestors de recursos (SGE entre ells), suport per a tot tipus de llibreries de paral·lelització (MPIch, OpenMP, Orte, ...), i té diverses característiques que el fan idoni per a gestionar un clúster:

- Creació de clústers dins de la xarxa:
  Permet definir de manera molt senzilla el conjunt de màquines que pertanyen a un clúster, permetent fins i tot jerarquies dins d'ell.

- Diferenciació de nodes:
  ROCKS clúster diferencia inicialment entre dos tipus de nodes: els nodes d'accés, o front-end, i els nodes de càlcul. El sistema requereix d'un node front-end el qual serà el node principal, de control, configuració i distribuïdor de la càrrega entre els nodes de càlcul, a la vegada que actua com a porta d'entrada dels usuaris al sistema.

- Imatges:
  Rocks-cluster parteix de la base que el software i sistema operatiu de totes les màquines serà el mateix. Degut a això, et permet generar una imatge del S.O. seleccionant els paquets i opcions que vulguem, i genera a partir d'ells una imatge que serà instal·lada a totes les màquines del clúster quan aquestes es reiniciïn (sigui en cas de caiguda, substitució, instal·lació d'una nova màquina, etc.).

- Manteniment dels fitxers de configuració:
  Rocks-cluster fa servir el sistema 411 per tal de mantenir actualitzats diversos fitxers de configuració del sistema, com el /etc/hosts, /etc/autofs, ... a tots els servidors que composen el clúster, de tal manera que quan realitzem un canvi a la configuració del node principal, aquest queda reflectit automàticament a la resta de servidors que composen el clúster.

Veient totes aquestes característiques, escollirem ROCKS com a sistema operatiu, ja que, si bé no té tantes "facilitats" com altres distribucions de Linux (no hi ha disponibles gaires paquets específics per a ell, però a l'estar basat en CentOS, que a la vegada pot fer servir paquets de RedHat i Fedora), si que ens ofereix una sèrie de característiques que el fan especialment indicat per a simplificar l'administració del múltiples servidors que conformen el clúster.

Un altre avantatge que ofereix ROCKS és el graf de kickstart, el qual està format per un conjunt de nodes, cadascun dels quals defineix un paquet a instal·lar o acció a executar durant el procés d'instal·lació de les màquines, així com les arestes entre nodes. Utilitzant això, es defineixen uns nodes "inicials" (per defecte només "compute" i "server") els quals seguiran tot el graf, executant tots els nodes amb què es trobi connectat. Al generar una imatge de ROCKS, s'inclou aquest graf en ella i al reinstal·lar una màquina, s'executarà un o altre camí depenent de com hagi estat definida aquesta màquina al clúster (com a servidor o node de càlcul).

Pel que fa al clúster d'alta disponibilitat, no l'inclourem dins d'aquesta infraestructura d'imatges del sistema operatiu, i optarem per una distribució més comuna com Suse Linux Enterprise, de la qual en disposem llicències a la universitat i que permet utilitzar HA-Linux per a establir un sistema d'alta disponibilitat.

Usuaris

Necessitem que tots els nodes del clúster comparteixin els mateixos usuaris, siguin nodes de càlcul, o part del clúster d'alta disponibilitat (contindrà alguns serveis que necessitaran autenticació d'usuari). Per tal de resoldre aquest problema, Rocks inclou de

sèrie una solució efectiva: el *daemon* 411, encarregat de copiar determinats fitxers de configuració des d'un servidor (front-end) als seus clients (nodes de càlcul). Aquest *daemon*, en la seva configuració inicial, comparteix ja els fitxers `/etc/passwd`, `/etc/group` i `/etc/shadow`, de manera que, efectivament, qualsevol canvi que es produeixi sobre els usuaris i grups es propagarà pels nodes de càlcul.

Aquesta solució, però, presenta el problema de que no s'estan actualitzant els usuaris del clúster d'alta disponibilitat. Això ho podríem solucionar simplement configurant aquests nodes com a clients del 411, però tenim també una alternativa, que consisteix en l'ús d'un servidor LDAP per a gestionar els usuaris de tot el sistema. Tot i que a efectes pràctics, el resultat és el mateix, l'ús de LDAP ens permet afegir al sistema informació addicional sobre els usuaris, com per exemple, el nom, l'adreça de correu electrònic, o el número d'empleat.

Per altra banda, pel que fa a la organització dels usuaris, tots els usuaris tindran un grup propi per al seu usuari, el qual serà el grup per defecte de tota l'estructura de directoris del seu home. Tots els usuaris formaran part també d'un grup comú sota el qual estaran aquells directoris que continguin elements d'ús comú per a tots ells. Finalment, hi haurà la possibilitat de crear grups relacionats als projectes del departament, als quals s'hi assignarà un subconjunt limitat d'usuaris, i que els permetrà compartir dades de manera més o menys privada entre ells.

Monitorització

Rocks-cluster inclou diversos sistemes de monitorització del sistema, com *Ganglia* i *Logwatch*, els quals s'encarreguen de mostrar-nos l'estat del sistema i advertir-nos en cas de problemes. Addicionalment, al departament disposem d'un servidor *Nagios* (software dedicat a monitoritzar diversos paràmetres d'un conjunt de hosts i a avisar en cas d'incidències), al qual hi connectarem tots els nodes del clúster.

Especificació del sistema

Tenint en compte totes les decisions que hem pres, la proposta final del sistema serà la següent:

Estructura del sistema

Optarem per una estructura de tipus clúster "*High-Throughput*" per al càlcul, conjuntament amb un petit clúster d'alta disponibilitat per a mantenir els serveis imprescindibles.

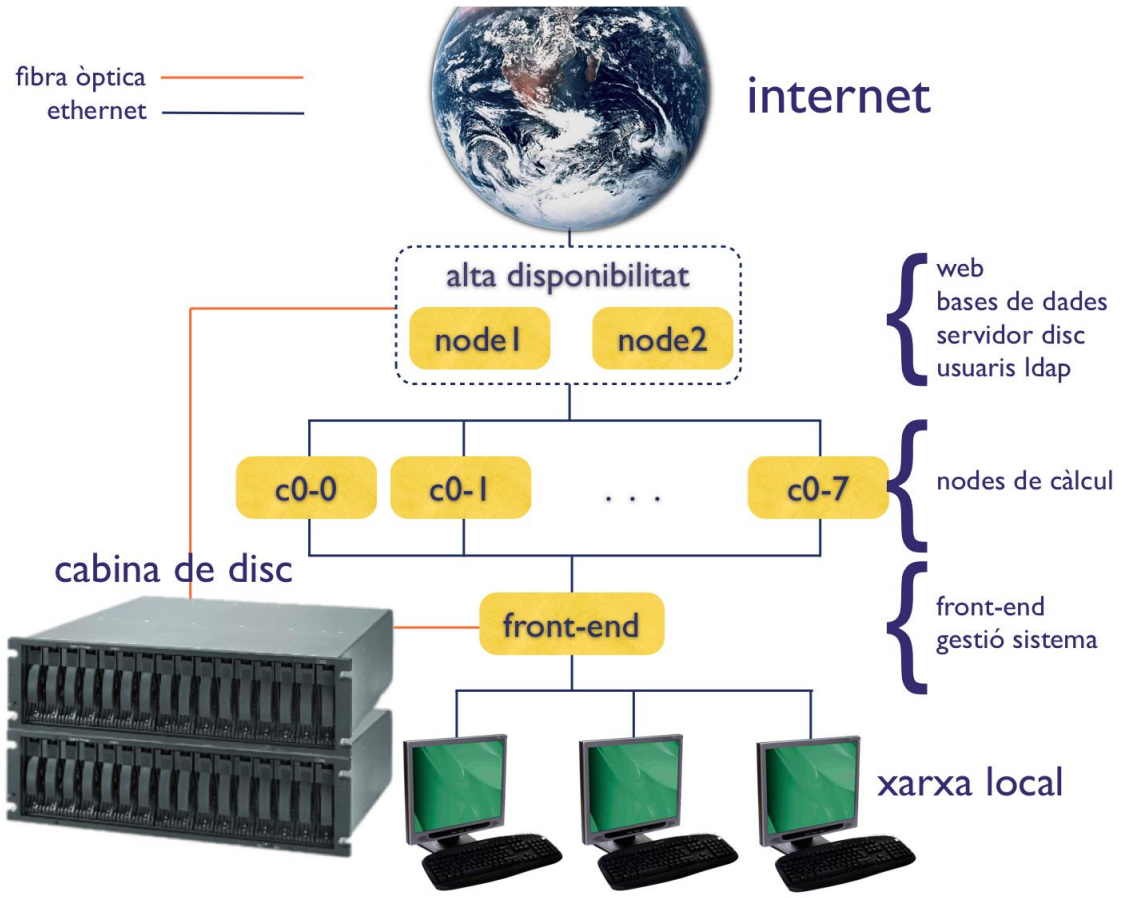

Figura 4.1 – Esquema de l'estructura proposada

Maquinari

Finalment el pressupost acceptat pel sistema estava composat pel següent:

Disc:

- Cabina de disc SAN: IBM Express DS4200 Model 7V.
- Doble font d'alimentació, doble controladora de Fibre Channel i llicència del software de gestió "IBM System Storage DS40002/FAStT Storage Manager 10".
- 11 discs Express DS4200 500GB 7.2K SATA EV-DDM HDD.

Xassís de blades:

- IBM eServer BladeCenter(tm) H Chassis.
- Doble font d'alimentació de 2900W i balancejador de càrrega DPI 63amp/250V Front-end PDU with IEC 309 2P+Gnd.
- 2 switchs Ethernet 1Gbps Nortel Networks Layer 2/3 Copper GbE Switch Module for BladeCenter.
- 2 switchs de fibra QLogic(R) 10-port 4 Gb SAN Switch Module BladeCenter.

11 blades consistents cadascun d'ells en:

- HS21 XM
- 2 processadors Xeon Quad Core E5345 2.33GHz/1333MHz/8MB L2, 2x512MB, O/Bay SAS
- 9 GB de memòria PC2-5300 CL5 ECC DDR2 Chipkill FBDIMM
- 1 disc Express IBM 73.4GB 10K SFF SAS HDD
- 1 tarja de KVM Concurrrent KVM Feature Card (StFF) for IBM BladeCenter
- 1 tarja de fibra QLogic 4Gb SFF Fibre Channel Expansion Card for IBM eServer BladeCenter

El cost total del sistema ha estat de 92.050,55 Euros. Podreu trobar el pressupost a l'*Annex I, pressupostos*.

Distribució de l'espai de disc

La distribució dels volums de disc serà la següent:

Discs locals:

- Servidors del clúster d'alta disponibilitat:
  - partició per al sistema operatiu: 60 GB
  - partició swap: 10 GB
- Front-end i nodes de càlcul:
  - partició per al sistema operatiu: 8 GB
  - partició per al /var: 8GB
  - partició swap: 10 GB
  - partició local (per a ser usada com a punt temporal en les execucions): 44GB

Cabina de discs:

- Volum *homes* (conté els homes, espai web i espai comú):
  Restringida per quotes molt limitades, de manera que podem assumir el backup del home i la web de tots els usuaris, més un espai extra comú per a instal·lació de programes i dades d'ús. Apliquem una política de quotes de 2 GB per usuari (amb uns 20 usuaris reals, però hem de preveure espai per a uns 50), necessitarem 100GB per als homes. Les quotes no afectaran a l'espai comú, ja que tot el que contingui aquest espai pertanyerà a l'usuari administrador (lliure de quotes).
    - Espai: 1 TB (100 GB homes + 900 GB comuns).
    - Quotes: 2GB / usuari.

- Volum *scratch*:
  Aquesta partició serà molt gran i no estarà restringida per quotes, tot i que disposarem de mesures de control que avisaran a tots els usuaris quan l'ocupació total del disc arribi al 80%.
    - Espai: 1 TB

- Volum *backups_desktop*:

  Limitada per quotes de 20 GB (per a 40 usuaris)

  - 800 GB
  - Quotes: 20 GB / usuari

- Volum *backups_homes*:

  La seva mida depèn del nombre d'usuaris del sistema i de les quotes de la partició homes:

  - 500 GB

- Volum *mysql*:

  La seva mida dependrà de l'ús que en facin els usuaris. D'entrada assignem un valor ampli però limitat, i en funció de l'ús que se'n faci l'ampliarem.

  - 150 GB + 50GB en una partició addicional per als logs binaris.

- Volum addicional:

  Els discs que queden contindran particions per altres menesters dels servidors del departament aliens al clúster.

Clúster d'alta disponibilitat

Format per dos dels blades, separats de l'estructura del clúster de càlcul (tot i que físicament dins del mateix xassís).

Sistema Operatiu:

- SuSE Linux Enterprise Server 10.2

Hostname:

- bhsrv1 i bhsrv2

Interfícies de xarxa:

- eth0: connectat a la xarxa externa, amb IP pròpia
  - bhsrv1.upf.edu 193.145.57.221

  - bhsrv2.upf.edu 193.145.57.222
- eth1: connectat a la xarxa privada (no enrutable)
  - bhsrv1.s.upf.edu 172.22.201.221
  - bhsrv2.s.upf.edu 172.22.201.222

Sistemes de fitxers:

Volum	Punt de muntatge	Espai
disc local	/	60 GB
swap	swap	10 GB
homes	/sp/fs/homes	1 TB
scratch	/sp/fs/scratch	1 TB
backup_desk	/sp/fs/backup_desk	800 GB
backup_homes	/sp/fs/backup_homes	500 GB
mysql_user	/sp/fs/mysql_user	150 GB
mysql_binlog	/sp/fs/mysql_binlog	50 GB

Serveis:

- **Alta disponibilitat**:
  - HA-Linux.
  - Heartbeat 2.1.4.

- Servidor **web** apache:
  - Apache 2.2.3.
  - Serveix com a webs personals acadèmiques els directoris situats dins del directori "public_html" al home de cada usuari.

- Servidor **ldap** OpenLDAP:
  - OpenLDAP 2.3.32.
  - Ordenació usuaris:
    - ou=Users,dc=upf,dc=edu
  - Ordenació grups:

        - ou=Groups,dc=upf,dc=edu

- Servidor **samba:**
    - Samba 3.0.32.
    - Accés només als homes dels usuaris.
    - Cada usuari només pot accedir al seu home.
    - Obert només per a la interfície interna (només s'hi podrà accedir des de la xarxa de la universitat).

- Servidor **mysql:**
    - Mysql 5.1.28-community.
    - Bases de dades en MySAM (en cas de que algun usuari ens demanés crear una BD que requerís moltes transaccions, li crearíem específicament en InnoDB).
    - Usuaris creats quan ho demanin.

- Servidor **NFS:**
    - Protocol NFSv3.
    - Particions a servir:
        - homes
        - scratch
        - backup_desk
    - Quotes.

Clúster de càlcul

Format per 1 front-end i 8 nodes de càlcul, tots ells dins del mateix xassís.

Sistema Operatiu:

- Rocks-cluster 5.0.

Tipus d'instància de node (al graph de kickstart):

- Front-end: Frontend.

- Nodes de càlcul: Compute.

Seqüència d'arrencada de BIOS:

- Front-end: disc.
- Nodes de càlcul: xarxa.

Hostname:

- front-end:
    - bhfront i cadaques (àlies).
- nodes de càlcul:
    - compute-0-0, ..., compute-0-7. Els noms els genera automàticament Rocks: El nom és el tipus de node, el primer número és el xassís on està, i el segon número l'ordre en què s'ha afegit al sistema.

Interfícies de xarxa:

- front-end:
    - eth0: connectat a la xarxa local del clúster.
        - bhfront.local - 172.22.205.1
    - eth1: connectat a la xarxa interna de la universitat.
        - bhfront.b.upf.edu - 172.22.202.200
- nodes de càlcul:
    - eth0: connectat a la xarxa local del clúster.
        - compute-x-x.local - 172.22.205.x
    - eth1: connectat a la xarxa privada de la universitat (no enrutable).
        - compute-x-x.s.upf.edu - 172.22.201.20x

Serveis:

La instal·lació de Rocks-cluster inclou tot el necessari per a tenir un clúster funcional: servidor web i mysql (d'ús intern per al sistema), ganglia (web de control dels nodes), sistema gestor de cues Sun Grid Engine, llibreries i compiladors MPI... Els únics canvis que cal realitzar al sistema són la configuració correcta d'alguns d'aquests serveis per a adaptar-los a les nostres necessitats, així com la modificació d'alguns paràmetres del graph de kickstart per tal d'automatitzar al màxim la instal·lació dels nodes de càlcul.

Usuaris:

Obtenció dels usuaris del sistema a partir del servidor d'OpenLDAP del clúster d'alta disponibilitat

Sistemes de fitxers:

Els servidors del clúster d'alta disponibilitat muntaran les particions de la següent manera:

Volum	Punt de muntatge	Espai
Disc local	/	8 GB
Disc local	/var	8 GB
Disc local	/state/partition1	34 GB
swap	swap	20 GB
homes (NFS)	/home/homes	1 TB
Scratch (NFS)	/home/scratch	1 TB
backup_desk (NFS)	/home/backup_desk	800 GB

Implementació del sistema

Instal·lació del hardware

Instal·larem tot el maquinari que formarà el nou sistema seguint les instruccions corresponents. Connectarem adequadament el xassís, fonts d'alimentació, ventiladors, switchs, blades, cabina i discs, obtenint una configuració final d'11 blades de dos processadors i 9GB de memòria, i una cabina de disc amb 11 discs de 500GB. Podreu trobar una visió més detallada a l'*Annex II, instal·lació del hardware*.

Connexions

La distribució de connexions de xarxa que utilitzarem per a connectar el sistema i integrar-lo a la nostra xarxa serà la següent (*figura 5.1*):

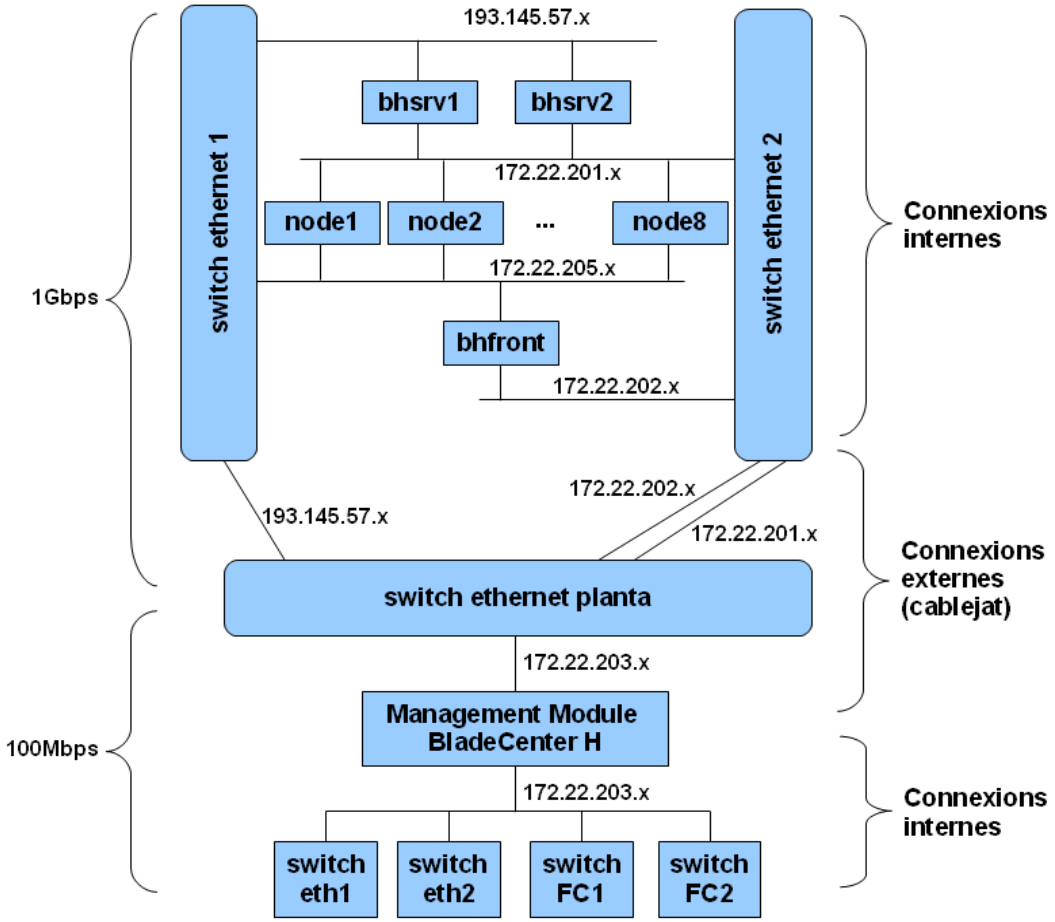

Figura 5.1 – Esquema de les connexions físiques tant internes com externes del xassís de blades i connexió al switch de planta principal del departament.

Com veiem, les connexions de xarxa dels blades es gestionaran dins del bus intern del xassís i només caldrà connectar 3 cables de xarxa (més el d'administració) des del xassís al switch del departament.

Pel que fa a la fibra òptica per a connectar el xassís a la cabina de disc SAN, la seva connexió serà més complicada a l'haver de mantenir i integrar les connexions de fibra que ja teníem al departament (*figura 5.2*).

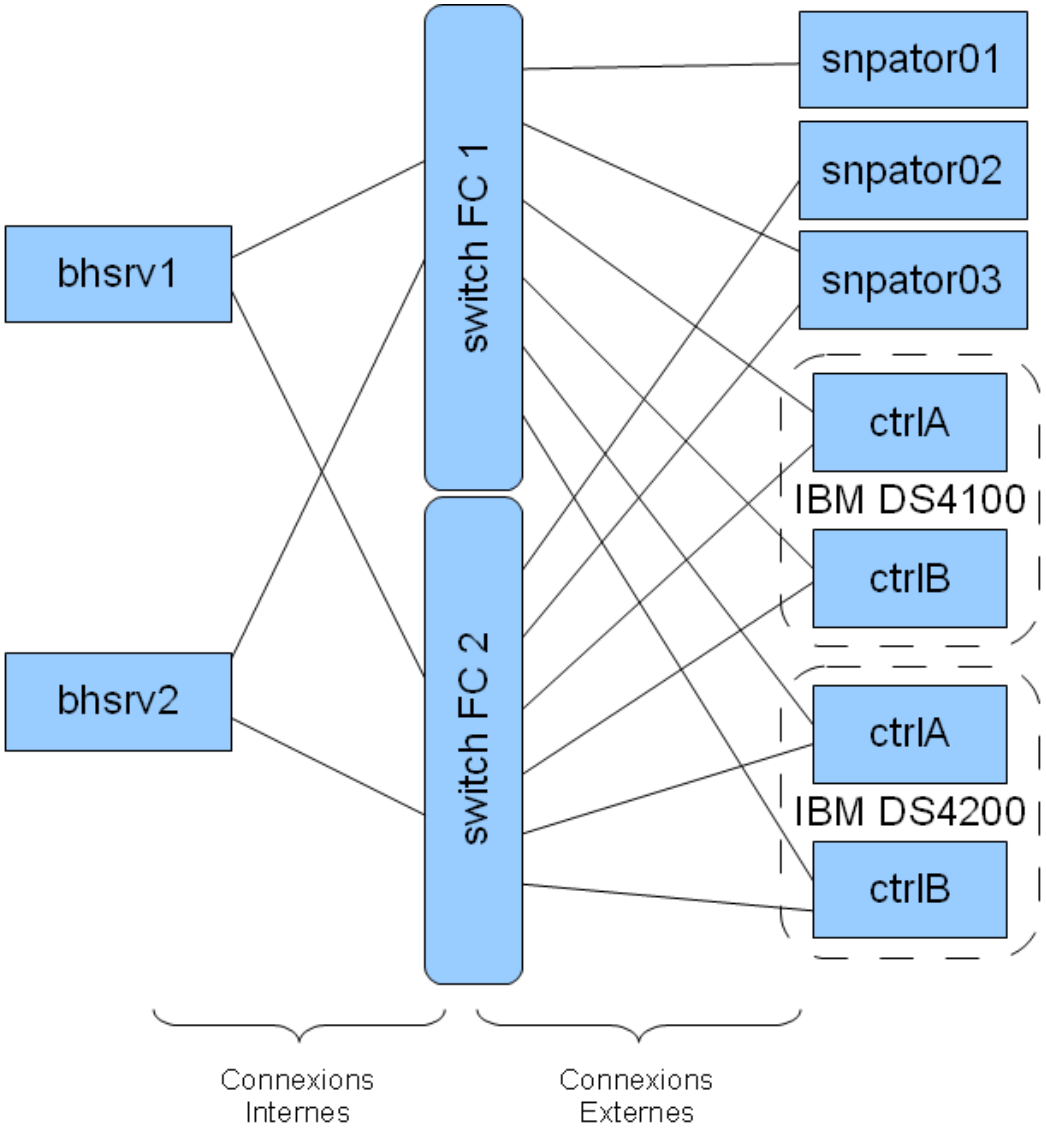

Figura 5.2 – Esquema de les connexions i switchs de fibra del departament després de la instal·lació dels nous xassís i cabina.

Distribució de disc

Pel que fa als discs que servirà la cabina, caldrà distribuir-los en grups de RAID. La cabina permet l'ús de múltiples configuracions de RAID, així com altres mesures de redundància i seguretat com discs de *Hot-Spare*: discs que "no fan res", però que estan a l'espera de que algun dels discs físics de la cabina falli per tal de prendre el seu lloc "en calent". Un cop passa això, el nou disc reconstrueix les dades que contenia l'antic a partir de la redundància emmagatzemada al RAID sense que el servei de disc es vegi interromput. Totes aquestes mesures de seguretat impliquen una pèrdua d'espai total de disc que podrem aprofitar. De les múltiples alternatives que tenim per a distribuir-los, finalment optarem per aquesta:

- Array "*homes*": 3 discs en RAID 5 (~1TB usable)
    - Logical drive: "*homes*" ~1TB
- Array "*scratch*": 3 discs en RAID 5 (~1TB usable)
    - Logical drive: "*scratch*" ~1TB
- Array "*data*": 4 discs en RAID 5 (~1.5TB usables)
    - Logical drive: "*backup_desk*" ~800GB
    - Logical drive: "*backup_home*" ~500GB
    - Logical drive: "*mysql*" ~150GB
    - Logical drive: "*mysql_binlog*" ~50GB
- Hot-Spare: 1 disc

D'aquesta manera, d'11 discs de 500GB (5,5TB bruts) que teníem, passem a disposar només de 3,5TB. A canvi d'aquesta pèrdua del 36% de disc, obtenim una doble redundància per a totes les nostres dades.

Clúster d'alta disponibilitat

Instal·lem a dos blades, els quals anomenarem bhsrv1 i bhsrv2, la distribució SuSE Linux Enterprise Server 10.2. Tots dos nodes compartiran configuracions idèntiques excepte pel que fa a la seva pròpia identitat, i seran els encarregats de mantenir diversos

serveis. A continuació detallarem breument aquests serveis, però podreu trobar informació més detallada a l'*Annex III, instal·lació del clúster d'alta disponibilitat*.

Sistema

La configuració del sistema quedarà de la següent manera:

Disc

Tots dos nodes tindran capacitat per a muntar els dispositius de disc connectats a través de fibra òptica des de la cabina SAN. A aquests discs s'hi accedirà a través de *multipath*, un servei encarregat d'abstraure tota la connexió real (targes, switchs, ports i controladores) utilitzada per a arribar físicament als discs, creant un dispositiu de disc virtual que ens assegurarà l'accés al disc independentment del camí seguit per la xarxa de fibra.

Aquests volums seran gestionats a través del LVM (*Logical Volume Manager*) de Linux, un sistema que permet virtualitzar els volums lògics de disc respecte als seus dispositius físics. Això ens permet, entre d'altres coses, tenir múltiples sistemes de fitxers en un sol volum de disc, o un sol sistema de fitxers en múltiples discs. Gràcies a aquest sistema podrem ampliar la mida dels volums lògics amb molta facilitat i molt més ràpid que si haguéssim d'afegir "físicament" un nou volum al raid dels discs de la cabina, i també ens dóna molta flexibilitat al permetre'ns, per exemple, ampliar una partició amb l'espai que no fem servir d'una altra.

Per a fer això, només caldrà definir un *Volume Group*, assignar-hi un conjunt de dispositius de disc físics que passaran al seu *pool* de blocs de disc, i definir els *Volums Lògics* que voldrem utilitzar (*figura 5.3*). Aquests volums lògics seran visibles des de Linux dins del directori `/dev`, i es comportaran com si es tractés d'un disc qualsevol. Addicionalment, la configuració del LVM queda emmagatzemada dins del disc físic, fent-la immediatament visible per a tots dos nodes del clúster d'alta disponibilitat.

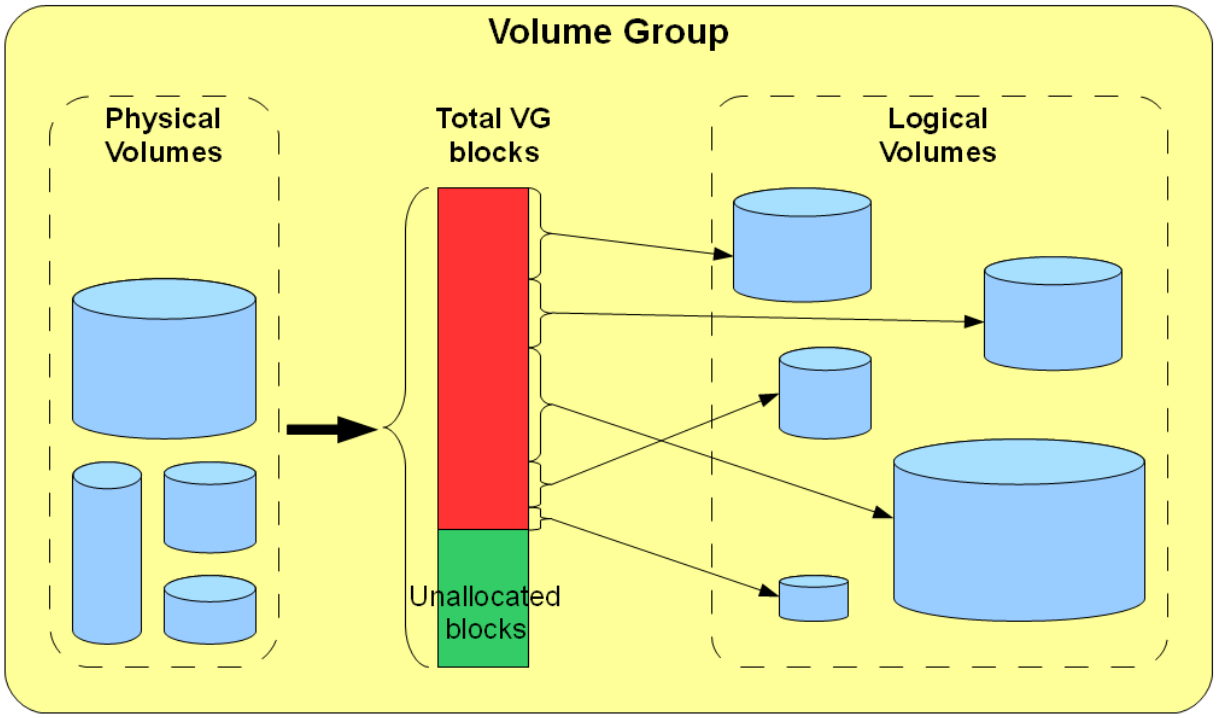

Figura 5.3 – Esquema de funcionament del Logical Volume Manager.

Un cop definits els volums lògics de disc de totes les particions que necessitem, caldrà definir el sistema de fitxers que els gestionarà. D'entre tot el gran ventall de sistemes de fitxers existents, optarem per un sistema de fitxers de propòsit general, ja que, si bé existeixen alguns dedicats a obtenir un màxim rendiment al tractar amb fitxers petits o grans, el nostre sistema inclourà dades de tots dos tipus dins d'una mateixa partició. Fins i tot en el cas de la partició destinada mysql ens trobarem amb fitxers de mida mitjana, quan seria un bon candidat a utilitzar un sistema específic per a fitxers grans com per exemple *xfs*.

Sabent això, configurarem tots els *filesystems* com a *ext3*, ja que és el sistema de fitxers de propòsit general que ens ha demostrat tenir una major estabilitat en cas de caiguda inesperada del sistema, desmuntatge forçat d'una partició, etc., sobretot comparant-lo amb *reiserfs*, el qual va ser descartat pels propis creadors de la distribució[3].

Finalment, configurarem les quotes de les particions *homes* i *backup_desk*, configurant-les inicialment a 2 i 20 GB per usuari, respectivament, però podent-se modificar segons evolucioni l'ús del sistema.

Estructura de directoris

Totes les particions seran muntades dins del directori `/sp/fs` dels nodes d'alta disponibilitat. Les particions dels usuaris (*homes*, *scratch* i *backup_desk*), però, seran muntades pels nodes de càlcul a través de NFS dins del directori home. Aquestes particions, a més a més, estaran entrellaçades entre elles mitjançant links (el home de l'usuari dins de la partició *homes* tindrà enllaços als directoris personals de l'usuari dins de les particions *scratch* i *backup_desk*). Degut a això, caldrà que mantinguem la coherència d'aquesta estructura de particions i enllaços per tot el sistema.

Serveis

Els serveis que mantindrà el clúster d'alta disponibilitat es configuraran de la següent manera:

NFS

De les 5 particions que muntaran en alta disponibilitat els nodes (*homes*, *scratch*, *backup_desk*, *mysql* i *mysql_binlog*), 3 seran servides via NFS a tots els nodes de clúster de càlcul (*homes*, *scratch* i *backup_desk*). Per a dur a terme això, configurarem el servei `nfsd`. Inicialment configurarem el servei per a servir totes tres particions, utilitzant el protocol nfs3 i els paràmetres de configuració estàndards. Pel que fa a la seguretat del servei, el configurarem de manera que només sigui accessible des de les IPs dels nodes del clúster de càlcul que l'hauran de muntar, així com del clúster d'alta disponibilitat. Durant l'etapa d'avaluació i test del sistema provarem el funcionament dels paràmetres que poden afectar al rendiment.

Samba

Mantindrem també en alta disponibilitat el servei de Samba que permeti accedir a les dades del clúster des de la xarxa de despatxos. Per a configurar el servei, limitarem l'accés només a aquelles màquines que arribin a través del proxy de la universitat (per a assegurar l'ús exclusiu des de la xarxa interna, i des de fora utilitzant la VPN de la

[3] Shankland, Stephen (2006-10-12). "Novell makes file storage software shift". *Business Tech* (cnet). http://news.com.com/Novell+makes+file-storage+software+shift/2100-1016_3-6125509.html.

universitat). Configurarem el servei per a que els usuaris només puguin accedir al seu propi home, i caldrà que creem 3 serveis diferents per a poder donar accés independent a cadascuna particions de dades dels usuaris (per compatibilitat entre Windows, MacOS i Linux). També caldrà crear un script de petició d'espai lliure al disc que respongui adequant les mides a la quota de l'usuari que realitza la petició.

Web

Configurarem el servidor web Apache. El seu objectiu serà servir les webs acadèmiques dels usuaris així com d'altres webs del departament. Per a les webs dels usuaris utilitzarem el mòdul d'apache `mod_userdir`, el qual farà accessible el directori `public_html` dins del home de cada usuari mitjançant peticions web de la forma "`http://domini/~username`". El clúster d'alta disponibilitat disposarà també d'una IP flotant, a la qual respondrà el domini `bhusers.upf.edu`. Aquesta IP serà utilitzada pel servei NFS i també per aquest servei web, el qual l'utilitzarà com a principal *virtualhost*. Per tal d'assegurar el correcte funcionament del servei en cas de canvi de host.

Usuaris

Instal·larem un servidor OpenLDAP replicat a tots dos nodes el qual servirà la informació de tots els usuaris del sistema, tant del clúster d'alta disponibilitat com de càlcul. El servidor el configurarem en alta disponibilitat, però enlloc de duplicar-lo als dos nodes i migrar el servei juntament amb la resta, crearem una configuració replicada de tipus *master-slave* entre les bases de dades LDAP dels dos servidors. Els usuaris del sistema quedaran desats sota l'esquema LDAP "`ou=Users,dc=upf,dc=edu`", i els grups sota "`ou=Groups,dc=upf,dc=edu`". Tot usuari tindrà el seu propi grup i tots els usuaris pertanyeran al grup comú "*bioevo*". També gestionarem la creació de grups per a projectes en què múltiples usuaris necessitin compartir informació. Addicionalment a tot això, crearem tot un conjunt de scripts per a gestionar les altes, baixes i modificacions d'usuaris i grups del sistema.

Base de dades

Utilitzarem MySQL com a servidor de base de dades. Distribuirem les dades en dues particions, una per les dades i l'altra per als logs binaris, per a millorar la seguretat de

les dades en cas de corrupció o problemes en un dels discs. L'objectiu d'aquest sistema es emmagatzemar només les bases de dades dels usuaris i, potser, la d'algun CMS en cas que un usuari decideixi fer-ne servir a la seva web personal. Inicialment farem servir un format de base de dades sense suport a transaccions com MySAM. En cas de que en algun moment fessin falta les transaccions, estudiaríem el cas concret per a passar a un format com InnoDB. Pel que fa als usuaris, com el servei de base de dades no serà d'ús molt estès, gestionarem els usuaris manualment.

Heartbeat

Per tal de gestionar el sistema d'alta disponibilitat, utilitzarem *Heartbeat*, part del projecte *HA-Linux*. Heartbeat consisteix en un sistema de control d'alta disponibilitat que permet definir diversos recursos, els quals monitoritzarà per a mantenir-los sempre actius en algun dels nodes en cas de que succeís qualsevol tipus d'incidència. Aquests recursos es podran definir individualment o en grup, dins del qual podrem assignar-los un ordre de tal manera que els processos s'iniciaran o pararan seguint una determinada seqüència, de manera molt similar als scripts de `init.d` d'arrencada del sistema.

Per al nostre sistema, un cop configurats tots dos nodes i les seves interfícies de comunicació, definirem dos grups de serveis:

- Servei de disc:
  Conjunt de serveis necessaris per a oferir el disc per NFS al clúster de càlcul. Per coherència de l'estructura de directoris, haurem de servir tots els volums en un únic bloc. La seqüència d'arrencada dels serveis que formaran aquest grup serà la següent:
  - Muntatge físic dels discs.
    - *Homes*.
    - *Scratch*.
    - *Backup_desk*.
  - Servei NFS.
    - *Nfsd*.
    - *Idmapd*.
    - *Rquotad*.

        - *Portmapper*.
    - IP flotant.
        - Pública.
        - Privada.
    - Servei Samba.
    - Servei Apache.
    - Reinici del Firewall per a adaptar-se als ports dinàmics del portmapper.

- Servei de base de dades:

  Engloba el conjunt d'elements necessaris per a mantenir el servei de base de dades:
    - Muntatge físic de les particions.
        - *Mysql*.
        - *Mysql_binlog*.
    - Servei mysql.
    - IP flotant del servei.
        - Pública.
        - Privada.

Un cop definits els serveis, els configurarem de tal manera que, tot i que tots dos serveis siguin capaços de funcionar a tots dos nodes, tinguin preferència a trobar-se en nodes diferents per a distribuir la càrrega.

Backup del sistema

Realitzarem la copia de seguretat del nostre sistema des del servidor de backup *Tívoli* que ja disposem al departament. Degut a les dimensions de disc del nostre sistema, només podrem fer backup de la partició homes. El nostre servidor primerament en farà una còpia de seguretat del disc a una partició externa (en aquest cas, el volum backup_homes que hem creat al definir els volums de disc de la cabina), i, posteriorment, en comprimirà les dades i les emmagatzemarà en cintes de backup utilitzant el robot de cintes del departament. La política de backup que definirem serà de backups incrementals diaris, emmagatzemant també 4 còpies setmanals i 3 mensuals.

Clúster de càlcul

Per a la instal·lació del clúster de càlcul utilitzarem la distribució de Linux Rocks-cluster 5.0. D'entre els múltiples paquets que aquesta inclou instal·larem aquells que contenen serveis interessants pel sistema, com són *ganglia* (el software de monitorització del sistema), *hpc* (les llibreries i programes de paral·lelització i càlcul d'alt rendiment), *java*, *intel* (compiladors) i *sge* (el software de gestió de cues i recursos Sun Grid Engine). Per a una informació més detallada de la instal·lació, podeu consultar l'*Annex IV, instal·lació i configuració del clúster de càlcul*.

Un cop acabada la instal·lació del sistema al node que actuarà com a front-end, i a la vegada de distribuïdor d'imatges dels nodes de càlcul, procedirem a modificar-ne la configuració per a adaptar-la a les necessitats concretes del nostre sistema. Els principals canvis que caldrà realitzar són:

Usuaris

Obtindrem els usuaris del servidor LDAP, el qual es troba replicat als nodes del clúster d'alta disponibilitat. Configurarem el sistema per a que accedeixi principalment al que actua com a *master*, i en cas que no estigui disponible, accedeixi a *l'esclau*.

Disc

Tots els nodes muntaran les tres particions de disc servides el clúster d'alta disponibilitat mitjançant NFS dins del directori `/home`. Els nodes de càlcul podran accedir al clúster d'alta disponibilitat directament a través de la xarxa 201, mentre que el front-end accedirà a través de la xarxa 202. Degut a aquesta diferència d'interfícies, caldrà modificar la configuració del client nfs per a permetre'n l'accés.

Llibreries compartides

Per tal de minimitzar les tasques de manteniment del sistema, instal·larem tots els programes en un punt d'accés comú a tots els nodes, dins del directori compartit `/homes/aplic`. Alguns dels programes que caldrà instal·lar, però, necessitaran

llibreries que no es troben al sistema. Per a evitar haver instal·lar-les a cada node, optarem per a deixar-les també en un directori accessible des de tots els nodes i modificarem la configuració de l'accés a llibreries (`/etc/ld.so.conf`) per a fer que els nodes de càlcul les incorporin. Addicionalment, crearem scripts per a gestionar, mantenir i actualitzar el llistat de llibreries dinàmiques de tots els nodes.

Serveis de xarxa

Configurarem alguns dels serveis de xarxa necessaris per a que s'adaptin a les configuracions del nostre sistema, com per exemple *postfix* (correu) o *ntp* (sincronització de rellotges).

Monitorització del sistema

Controlarem l'estat del sistema utilitzant dos serveis, *Ganglia* i *Nagios*:

Ganglia és un aplicatiu que s'inclou ja amb la instal·lació de rocks i que ens permet visualitzar, a través d'una interfície web gràfiques de l'estat del disc, memòria, processador i xarxa de tots els nodes del sistema. La instal·lació inicial del servei és plenament funcional, però la modificarem per a afegir també a la monitorització els nodes del clúster d'alta disponibilitat.

Per altra banda, *Nagios* és un sistema que monitoritza l'estat del node mitjançant l'execució de diversos scripts que ens informen de l'estat d'un determinat paràmetre. Aquests scripts són plenament configurables i envien la informació a un node servidor del que en el nostre cas ja disposàvem al departament.

Amb la configuració del sistema ja adaptada i definida, creem i instal·lem la imatge del sistema operatiu a tots els nodes. Un cop acabada la instal·lació, només a faltarà configurar el sistema gestor de cues del clúster.

Configuració de Sun Grid Engine

Al finalitzar la instal·lació dels nodes, el sistema gestor de cues SGE (i per tant, el funcionament del clúster de càlcul) ja és plenament funcional, tot i que caldrà modificar-ne la configuració per a millorar el sistema. SGE posa a la nostra disposició multitud d'opcions de configuració, des de la creació de cues, gestió de càrrega, tiquets de consum de recursos, distribució per usuaris, projectes i departaments, paràmetres complexes, configuracions individuals per cada node de càlcul, etc. Com inicialment, tant nosaltres com els usuaris, desconeixem quina configuració s'adaptarà millor a les necessitats dels *jobs* (processos) que executaran els usuaris, i només podem especular, optarem per a posar el sistema en producció i adaptarem la configuració en diverses iteracions a mesura que vagi evolucionant l'ús del sistema que en fan els usuaris.

La configuració inicial de SGE consisteix en una única cua anomenada "all.q" que conté a tots els nodes de càlcul. Cada node podrà admetre fins a 8 jobs (un per core), i els usuaris es repartiran els jobs utilitzant una cua FIFO. L'única restricció que farà aquesta configuració bàsica és intentar equilibrar la càrrega dels nodes, distribuint els nous processos sempre a les màquines amb menor càrrega de CPU.

Avaluació i test

Un cop instal·lat i configurat el sistema, hem d'avaluar el funcionament i el seu rendiment. Per a això avaluarem per separat diversos elements crítics del sistema.

Clúster d'Alta Disponibilitat

Dos dels elements més crítics del sistema i que hem d'avaluar, es troben instal·lats al clúster d'alta disponibilitat: Heartbeat i el servei NFS

Heartbeat

El correcte funcionament de heartbeat dependrà, a part de la evident correctesa del funcionament del servei activat, la velocitat amb què es detecta la caiguda del servei i la seva restauració. En resum, el temps que estarem sense servei. Aquest funcionament el podem mesurar en dues dades: el temps que triga en detectar que un dels dos nodes ha caigut, i el temps que triga en activar els serveis.

Partint de la configuració inicial següent:

```
Keepalive:                    1000ms
warningalive:                 1500ms
deadtime:                     3
initialdeadtime:              12
default resource stickiness:  0
default resource failure stickiness:    0
cluster delay:                60s
batch límit:                  30
Default Action Timeout:       20s
Stop Orphan resources:        true
DC deadtime:                  10s
```

Els resultats que obtenim empíricament de l'ús del sistema de heartbeat són els següents:

Acció	Serveis afectats	Temps (seg)
reinici d'un node	mysql_user	462,83
reinici d'un node	homes, scratch, backup_desk, samba, apache	478,42
moure servei	mysql_user	9,8
moure servei	homes, scratch, backup_desk, samba, apache	25,4

Amb aquests resultats comprovem que el funcionament del sistema d'alta disponibilitat és correcte. Funciona de manera adequada i sense retards i detecta immediatament (dins dels límits d'avís determinats a la configuració) totes les incidències del sistema. Podem fer, però, diverses observacions sobre el seu estat:

- Veiem que el reinici d'un node, des de l'instant en què rep l'ordre d'aturar-se fins que tots els serveis tornen a estar operatius, és de vora 470 segons (quasi 8 minuts). Aquest temps hauria de ser molt més curt, però durant la seqüència d'inici, el node s'està força temps ocupat en la comprovació de les metadates del Logical Volume Manager. Això és degut a que el servidor revisa totes les particions que veu a través de fibra òptica, les vagi a muntar o no. Si reduíssim l'ús de LVM en les particions milloraríem aquest temps, però a canvi perdríem els avantatges que ens proporciona usar-los (principalment ampliar el volum dels discs de manera molt més flexible i ràpida). Tot i això, aquest apartat no serà crític, ja que, gràcies a heartbeat, mentre el node s'estigui reiniciant el servei és plenament funcional a l'altre node.

- El servei *homes* triga molt més en reiniciar-se al canviar de node, degut principalment a que munta tres volums (homes, scratch i backup_desk), així com diversos serveis (samba, nfs i apache).

Durant les proves de rendiment hem detectat un problema amb el sistema de heartbeat degut a que només tenim dos nodes en alta disponibilitat: En cas de fallada de xarxa tots dos assumeixen simultàniament que, al no poder comunicar-se amb l'altre node, l'altre ha caigut i ell és l'únic supervivent. Això porta a tots dos nodes a intentar muntar tots els serveis simultàniament, provocant diversos problemes com potencials corrupcions

de disc. Aquesta situació, però, és molt poc probable, ja que implicaria una fallada del switch o de les targes de xarxa, i això deixaria als nodes sense xarxa, i, per tant, sense poder proporcionar els serveis a la resta del sistema igualment.

Per a prevenir aquesta possible situació, a la propera aturada de manteniment que fem del sistema afegirem un nou node al clúster d'alta disponibilitat. Aquest node no caldrà que munti cap servei. De fet, una de les configuracions possibles és l'ús com tercer node de *Quòrum* de qualsevol màquina ja existent connectada a les xarxes dels dos nodes actuals. Aquest tercer node tindrà explícitament prohibit el muntatge de serveis, però estarà connectat per a donar constància de la seva presència. D'aquesta manera, a l'estar el clúster format per 3 nodes, en cas de caiguda de la xarxa o problemes similars, només el node que vegi més de la meitat dels nodes (en aquest cas 2) muntarà els serveis. L'altre, que només en veurà un (la màquina aïllada), aturarà els seus serveis. Això no impedirà que caiguin tots els serveis en cas de problemes de xarxa (switch), però sí que impedirà un possible escenari en que tots dos nodes els intenten muntar i fallen.

NFS

La configuració escollida inicialment per al servei NFS funciona i dóna servei correctament a tots els nodes del clúster. Partint d'aquesta configuració, analitzarem com afecten al rendiment del servei els diferents canvis que podem realitzar als paràmetres de configuració. Els paràmetres que poden afectar al rendiment són:

- Al servidor:
  - Nombre de **threads:**
    - 1-N. La documentació recomana utilitzar un per a cada "petició" de client; en el nostre cas 64. Augmentar el nombre de processos fins al nombre de cores del servidor (8 en aquest cas), teòricament augmentarà el rendiment del sistema a l'aprofitar completament la CPU del servidor, a no ser que hi hagi un coll d'ampolla en l'escriptura a disc. Augmentar el nombre de threads per sobre del nombre de cores teòricament millorarà el rendiment a l'evitar "canvis de context" en un mateix thread al canviar el procés al que està assignat.

- Versió del **protocol**:
  - 2, 3 o 4. La versió 2 la descartem ja que està desfasada i la documentació diu que només es manté per temes de compatibilitat amb sistemes antics. Respecte a les versions 3 i 4, no hi ha una clara preferència: Tot i que teòricament la versió 4 inclou millores de rendiment i seguretat, cap d'elles sembla ser aplicable en el nostre cas.

- **GSS**:
  - Sí/No: Activar o no GSS com a capa de seguretat utilitzant Kerberos. Com utilitzem una xarxa privada, no ens resulta necessari, però comprovarem el seu impacte en el rendiment.

- **Wdelay:**
  - wdelay/no_wdelay: S'activa per partició exportada. Les escriptures "físiques" al disc es retarden preveient la possible arribada d'una altra escriptura a continuació. El seu efecte sobre el rendiment és incert: en cas de tenir un únic procés escrivint un arxiu gran, és una clara millora, però si tenim fins a 64 processos escrivint simultàniament, una espera pot provocar una disminució del rendiment global.

- Al client:
  - Versió del **protocol**:
    - 2, 3 o 4: Tot i que el servidor ha d'especificar quines opcions accepta el servei, és el client qui decideix quina fa servir per a cada volum.

  - Protocol de **xarxa**:
    - TCP/UDP: TCP és un protocol "segur", que requereix de tres paquets per a establir una connexió. UDP és un protocol "insegur" que necessita un menor nombre de paquets. Teòricament UDP permet una major velocitat en la comunicació en xarxes amb molt trànsit, a costa de sacrificar certa fiabilitat en la transmissió de paquets.

- **Wsize/Rsize**:
  - Mida de les finestres de lectura i d'escriptura que configurarem al client. Es negocia entre el client i el servidor. El seu valor màxim el trobem definit al servidor a `/usr/src/linux/include/linux/nfsd/const.h` on la variable `NFSSVC_MAXBLKSIZE` la tenim definida a 32KB. El valor per defecte al muntar el volum és de 512 bytes. Teòricament, valors inferiors al màxim influiran no influiran negativament en el rendiment, però impediran obtenir-ne el màxim.

- A ambdós:
  - **MTU**:
    - Els nostres switch són compatibles amb *Jumbo Frame*, la qual cosa permet fer servir paquets de transmissió de 9000 bytes enlloc dels 1500 habituals. Si configurem les targes de xarxa tant als clients com al servidor per a acceptar-les, podrien accelerar la comunicació.

Per a fer les proves de rendiment, utilitzarem la següent metodologia:

- Engegar el servei `nfsd` al servidor utilitzant una combinació de paràmetres (mtu, wdelay/no_wdelay, gss, nombre de threads).
- Muntar el volum NFS a tots els clients que faci falta, amb la combinació de paràmetres corresponent (mtu, tcp/udp, versió de NFS, wsize, rsize).
- Utilitzar la comanda "`dd`" per a escriure al disc. Per a testar les escriptures llegirem de `/dev/zero` i escriurem al volum NFS. Per a les lectures llegirem un fitxer al volum NFS i ho enviarem a `/dev/null`. Utilitzarem diverses mides de bloc per a simular diversos tipus d'escriptures.
- Executar simultàniament diversos tests utilitzant SGE per a simular l'efecte de tenir múltiples processos treballant sobre el disc simultàniament.

Els resultats els obtindrem de l'output de la comanda "`dd`". La fiabilitat d'aquests, però, es veurà esbiaixada quan es tracti de diversos processos simultanis: tot i executar-se simultàniament, no tots els processos acabaran alhora, fent impossible calcular la

mitjana. Per a corregir això, utilitzarem les gràfiques que genera Ganglia sobre l'activitat de xarxa del servidor, les quals ens permetran veure l'activitat "global" del sistema.

Proves d'escriptura

Com podem veure *(taula 6.1*), trobem alguns punts sense valor marcats amb "-". Aquestes caselles indiquen punts on, degut a la "no-simultaneïtat" de les peticions d'escriptura, els resultats donen un valor irrealment alt (generalment per sobre de 90MB/s). Aquests resultats són falsos artefactes que no apareixen a les gràfiques d'ús de xarxa i ens ho trobarem en diversos casos. Per exemple, en el cas de les 4 escriptures simultànies amb 4 threads nfs, protocol TCP i mtu de 1500 (*figura 6.1*) i mtu 9000 (*figura 6.2*), o en el cas de les 24 escriptures simultànies amb 16 threads, mtu de 1500 i protocol TCP (*figura 6.3*) o protocol UDP (*figura 6.4*). Com podem veure en tots els casos, tot i que els valors obtinguts numèricament estan molt per sobre del normal, a la gràfica d'activitat de xarxa veiem perfectament com el seu ús entra dins dels valors esperats (~60MB/s).

Per altra banda, també podem veure com l'ús dels protocols TCP i UDP afecten al nombre de paquets necessaris per a la comunicació. Per exemple, en el cas de les 4 escriptures amb 4 threads i mtu 1500 podem veure com a l'utilitzar TCP (*figura 6.5*) estem generant moltíssims més paquets (entre el doble i deu vegades més) que si utilitzem UDP (*figura 6.6*).

Proves de lectura

Per contra, en les proves de rendiment de les lectures a disc (*taula 6.2*), ens trobem una molta millor qualitat de les dades. Els valors obtinguts són molt consistents, especialment quan més ens apropem al llindar de 32KB, apropant-se molt al límit teòric de la xarxa Gigabit. També podem observar els valors disparats dels resultats quan entra en joc la cache.

#threads nfsd	#simult anis	opcions	protocol	mtu	WSIZE													
LOCAL				BS=	512b	1K	1500	2K	4K	8K	9000	16K	24K	32K	56K	64K	128K	256K
					151,63	110,11		87,46	129,14	90,35		85,91	150,63	92,89	105,97	121,65	77,18	118,92
4	1	wdelay gss2	tcp	1500	56,30	27,00	26,93	38,85	56,44	65,37	65,46	65,91	66,05	66,16	66,37	66,27	66,32	66,27
4	1	wdelay gss2	tcp	9000	62,67	26,77	26,87	42,43	62,59	65,27	65,31	66,17	66,18	66,55	66,64	66,66	66,54	66,70
4	1	wdelay gss2	udp	1500	63,99	38,82	38,84	51,94	63,97	66,57	66,41	67,01	66,96	67,17	67,34	67,25	67,34	67,24
4	1	wdelay gss2	udp	9000	65,87	38,83	38,82	53,88	66,23	68,02	67,96	68,39	68,40	68,56	68,02	68,16	68,14	68,11
4	1	no_wdelay	tcp	1500	56,46	26,80	26,84	38,93	56,39	65,66	65,79	66,32	66,27	66,63	66,60	66,54	66,60	66,56
4	1	no_wdelay	tcp	9000	62,76	26,86	26,77	42,47	62,94	65,45	65,46	66,55	66,63	66,96	66,83	66,74	66,30	65,97
4	1	no_wdelay	udp	1500	64,21	38,95	38,98	52,11	64,29	66,74	66,80	67,24	67,27	67,71	67,53	67,65	67,66	67,49
4	1	no_wdelay	udp	9000	64,40	38,64	38,63	53,31	65,38	67,44	67,73	68,17	68,22	68,39	68,44	68,30	68,28	68,32
4	2	no_wdelay	tcp	1500	46,99	16,31	16,36	27,31	47,86	62,50	59,11	66,50	61,30	65,73	66,05	65,02	64,67	65,92
4	2	no_wdelay	tcp	9000	56,19	15,47	16,67	31,06	52,31	65,77	64,50	66,60	66,03	67,65	64,47	65,07	61,30	67,13
4	2	no_wdelay	udp	1500	56,66	20,09	17,69	35,30	55,35	61,68	64,56	62,63	64,90	64,14	62,50	65,72	64,75	62,37
4	2	no_wdelay	udp	9000	60,12	17,58	18,86	36,01	54,99	64,63	65,38	62,87	65,05	66,66	64,56	65,00	63,74	66,26
4	4	no_wdelay	tcp	1500	51,89	21,24	21,44	33,49	57,25	65,65	62,90	-	68,20	-	-	-	-	-
4	4	no_wdelay	tcp	9000	57,88	20,29	19,54	42,08	59,44	-	-	-	-	-	-	-	-	-
4	4	no_wdelay	udp	1500	57,81	22,59	23,52	39,73	56,72	64,53	63,90	-	-	60,60	63,50	64,70	67,45	64,70
4	4	no_wdelay	udp	9000	56,52	23,74	21,94	41,42	57,83	-	69,05	68,20	69,25	66,60	66,90	-	-	
4	24	no_wdelay	tcp	1500	26,50	1,03	0,74	14,44	26,16	41,23	41,80	62,27	61,87	80,49	78,05	77,85	78,05	77,77
4	24	no_wdelay	tcp	9000	28,28	0,85	0,99	21,34	27,56	55,78	57,73	73,57	76,05	84,73	83,49	83,91	83,88	83,74
4	24	no_wdelay	udp	1500	30,13	1,11	1,04	23,76	29,95	72,38	64,36	85,32	89,19	88,57	81,70	-	-	
4	24	no_wdelay	udp	9000	35,66	1,42	1,37	24,88	33,26	76,57	78,54	-	-	-	-	-	-	
4	24	wdelay	tcp	1500	48,38	21,40	20,56	29,89	44,95	57,48	51,77	61,43	63,36	72,19	78,99	78,50	80,74	78,45
4	24	wdelay	udp	1500	49,28	22,42	22,25	28,96	51,66	77,58	70,61	78,01	79,61	82,10	88,79	88,13	88,69	84,64
8	24	wdelay	tcp	1500	58,17	24,56	24,04	37,37	58,20	71,80	76,71	76,75	77,86	80,00	82,10	82,60	84,37	85,90
8	24	wdelay	udp	1500	66,26	26,48	25,75	36,43	65,05	83,46	82,05	-	-	-	-	-	-	
16	24	wdelay	tcp	1500	79,93	44,96	47,44	55,74	81,05	-	88,20	-	-	-	-	-	-	-
16	24	wdelay	udp	1500	85,60	45,85	46,44	64,53	86,28	-	89,80	-	-	-	-	-	-	-
64	24	wdelay	tcp	1500	29,60	54,40	58,31	47,28	51,00	41,80	64,13	61,15	49,05	74,30	-	72,35	58,10	70,30
64	24	wdelay	udp	1500	71,24	63,28	73,57	68,95	47,12	69,81	63,71	38,48	48,14	53,50	48,47	67,25	67,00	49,00

Taula 6.1 – Velocitat d'escriptura a disc en MB/s.

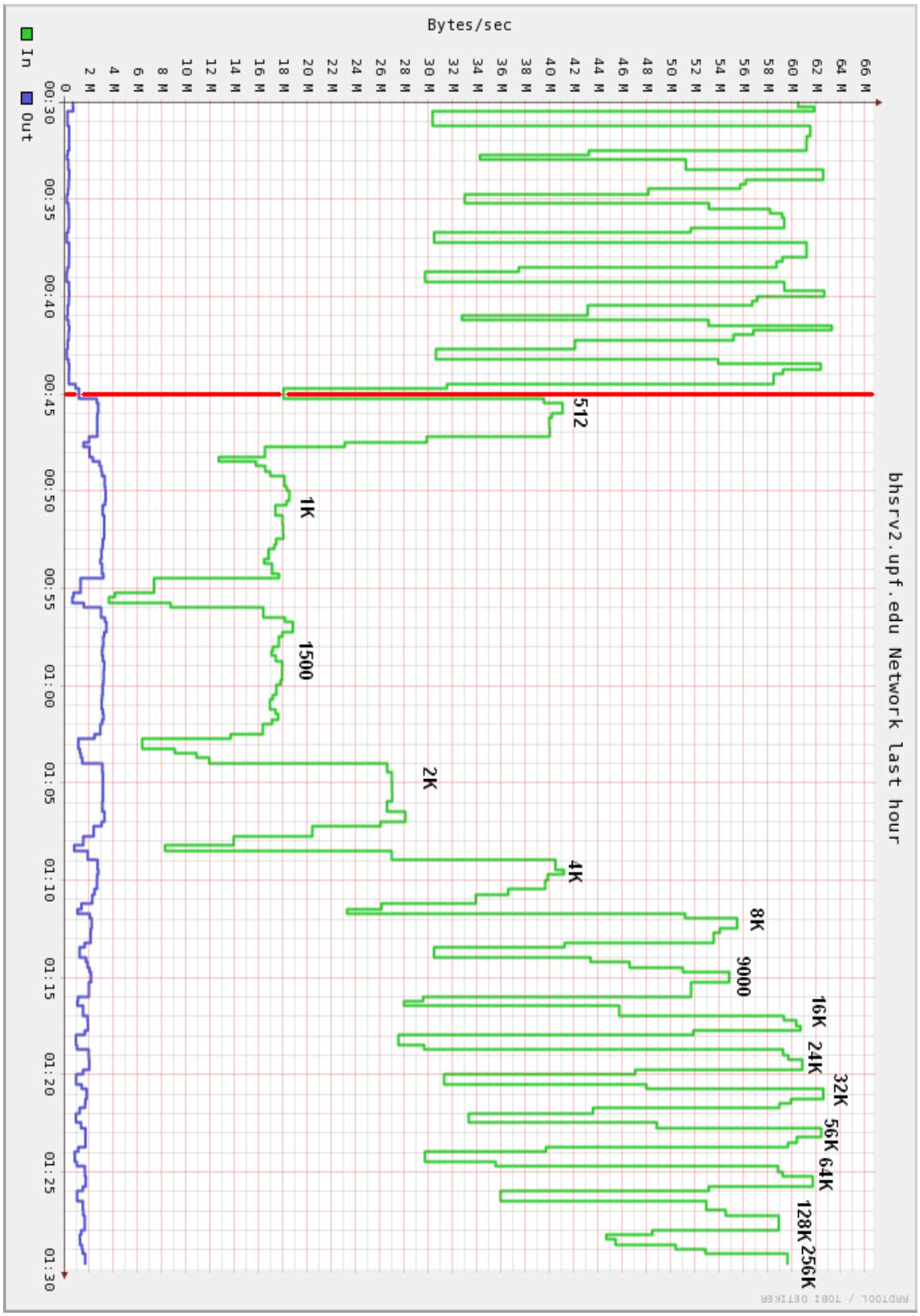

Figura 6.1 – Estat de la xarxa durant les proves d'escriptura de 4 escriptures simultànies amb 4 threads, nfs v3, TCP i mtu=1500.

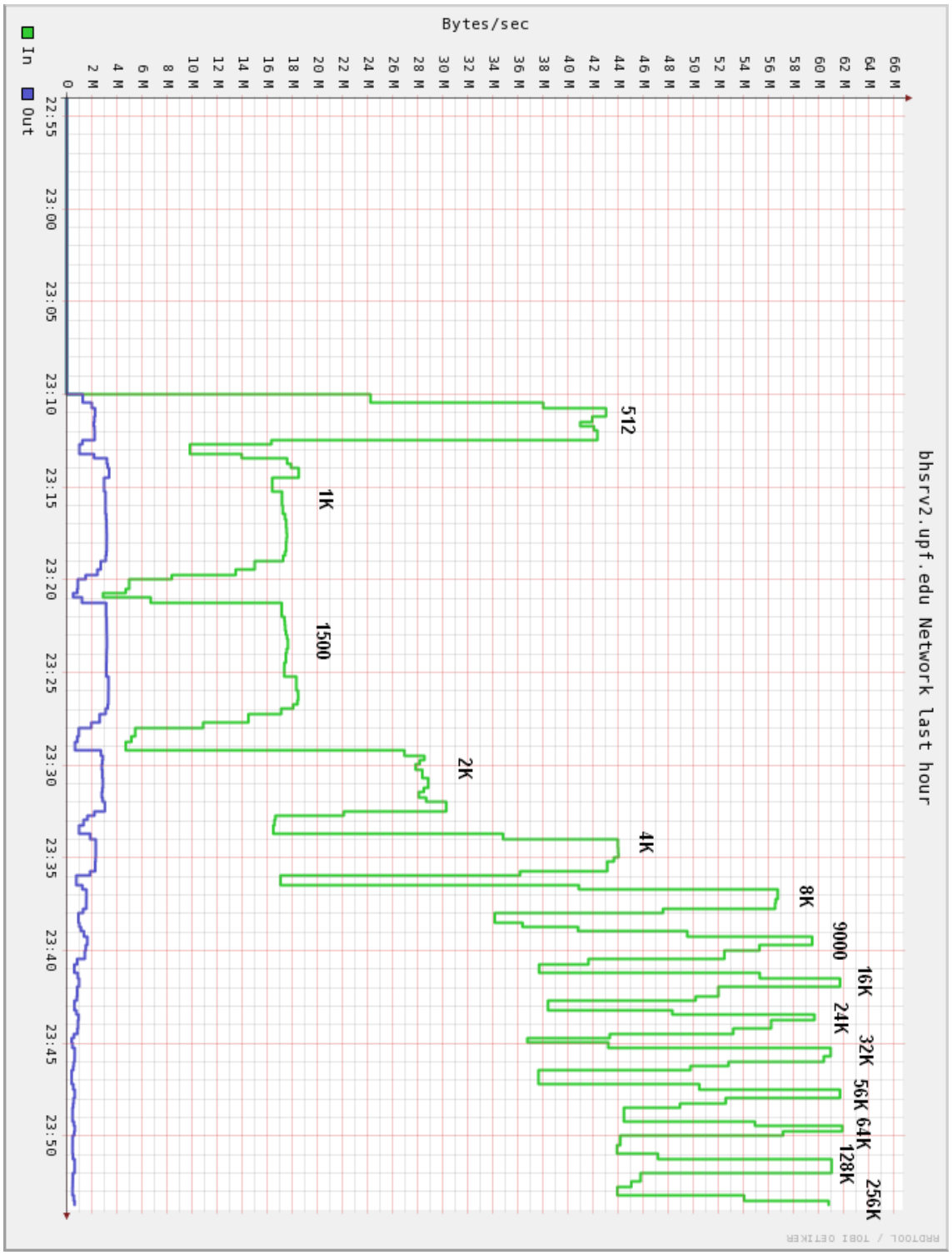

Figura 6.2 – Estat de la xarxa durant les proves d'escriptura de 4 escriptures simultànies, 4 threads, nfs v3, TCP i mtu=9000.

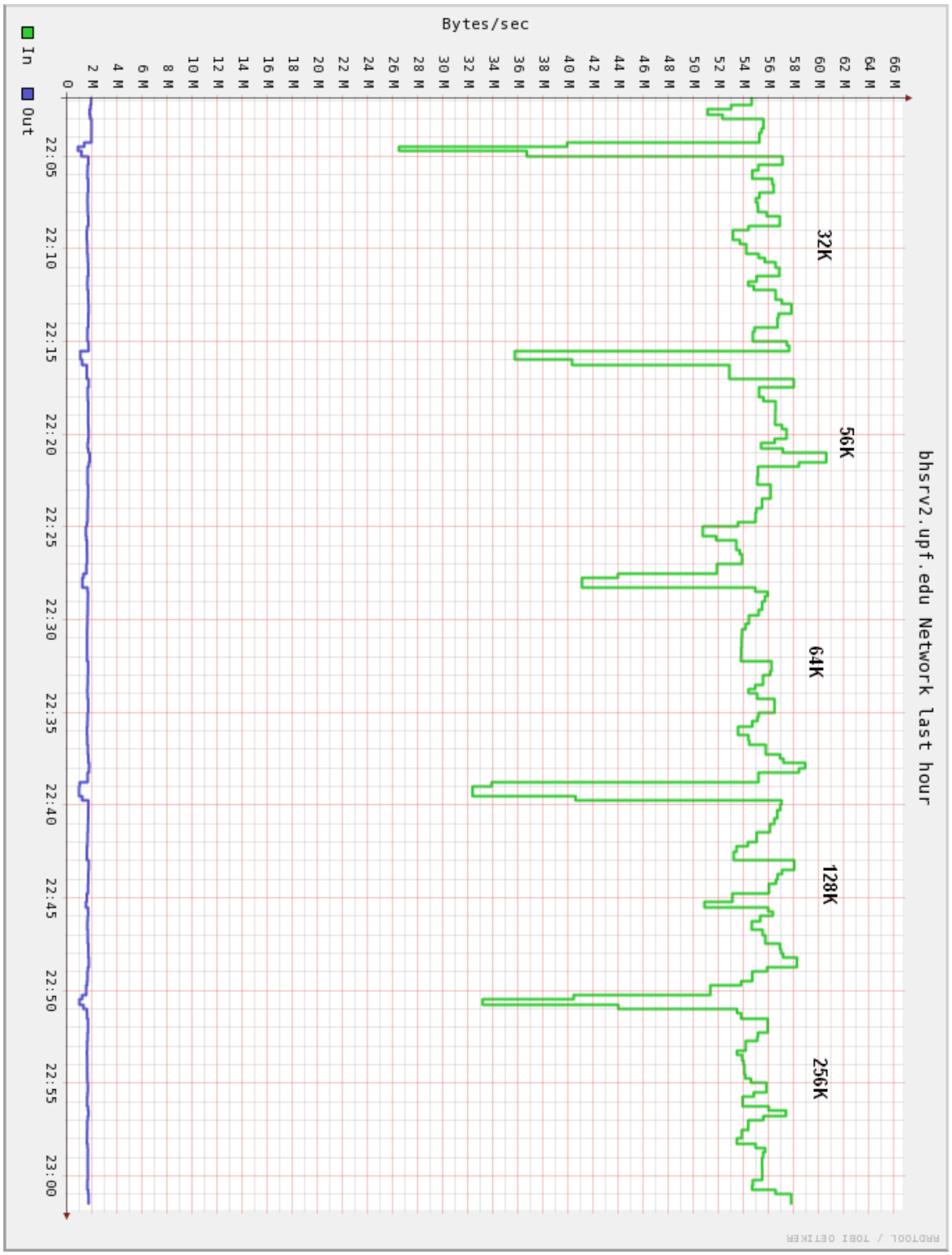

Figura 6.3 – Estat de la xarxa durant les proves d'escriptura de 24 escriptures simultànies amb 16 threads, nfs v3, TCP i mtu=1500.

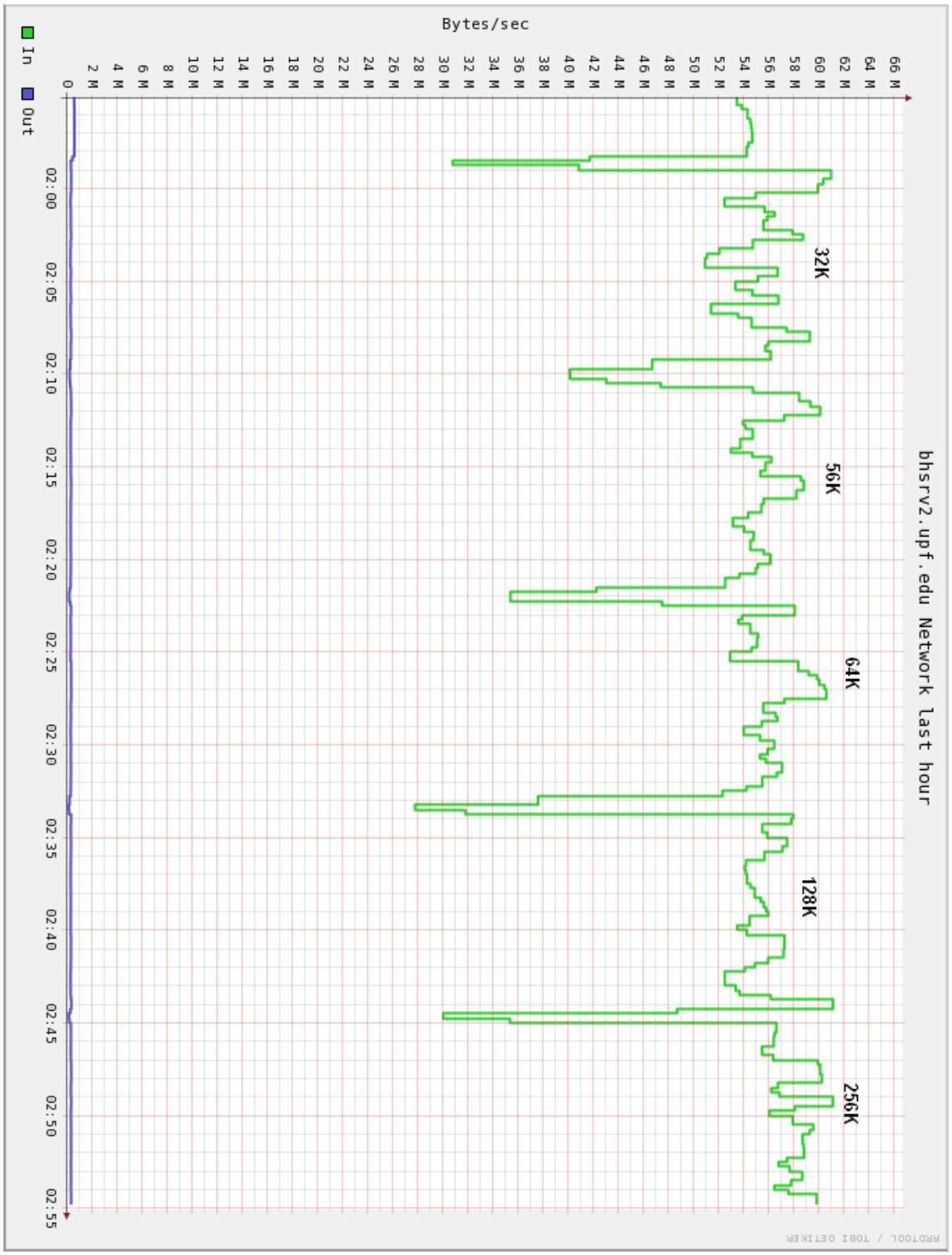

Figura 6.4 – Estat de la xarxa durant les proves d'escriptura de 24 escriptures amb 16 threads, nfs v3, UDP i mtu=1500.

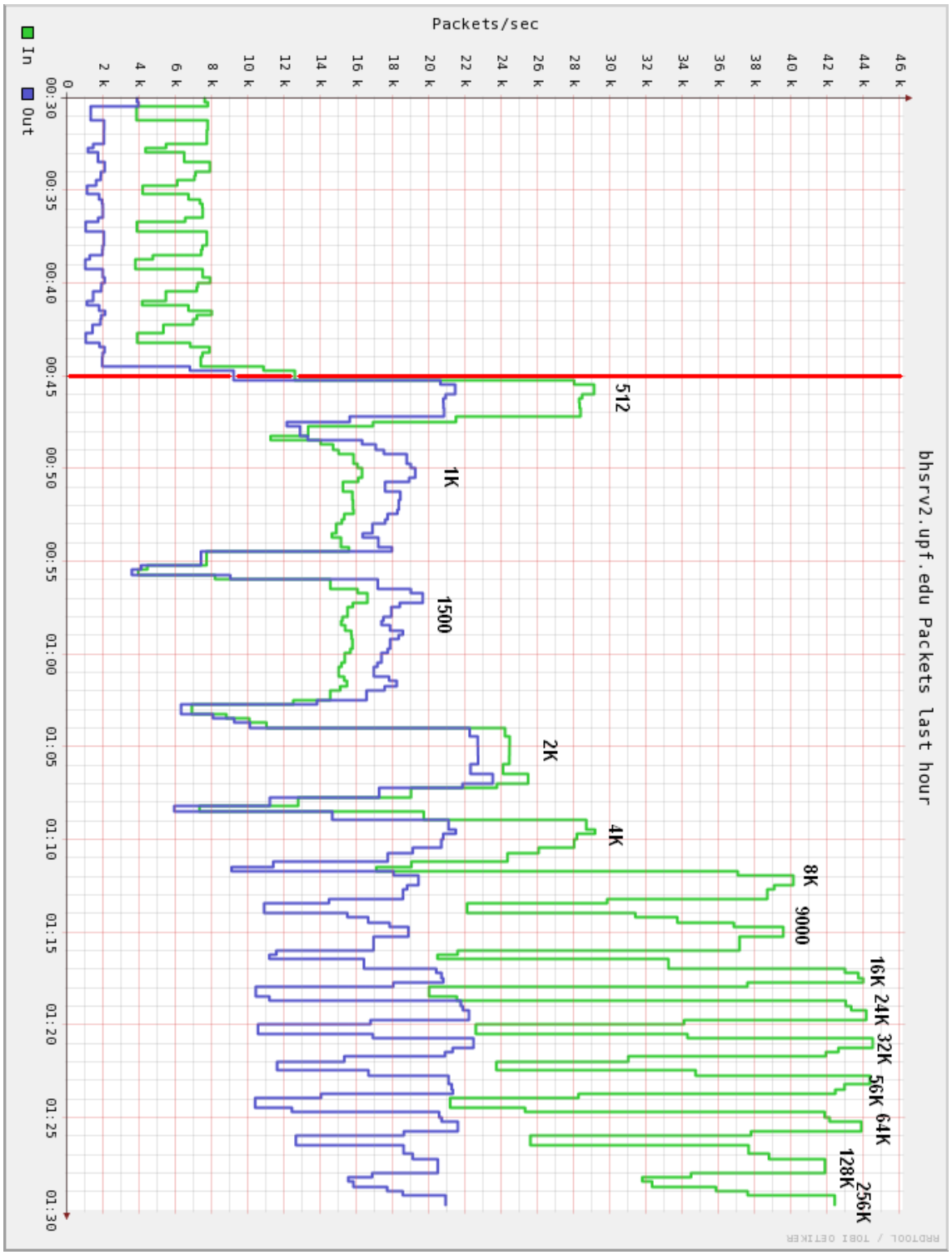

Figura 6.5 – Tasses de transmissió de paquets de la xarxa durant les proves de 4 escriptures simultànies amb 4 threads, nfs v3, TCP i mtu=1500.

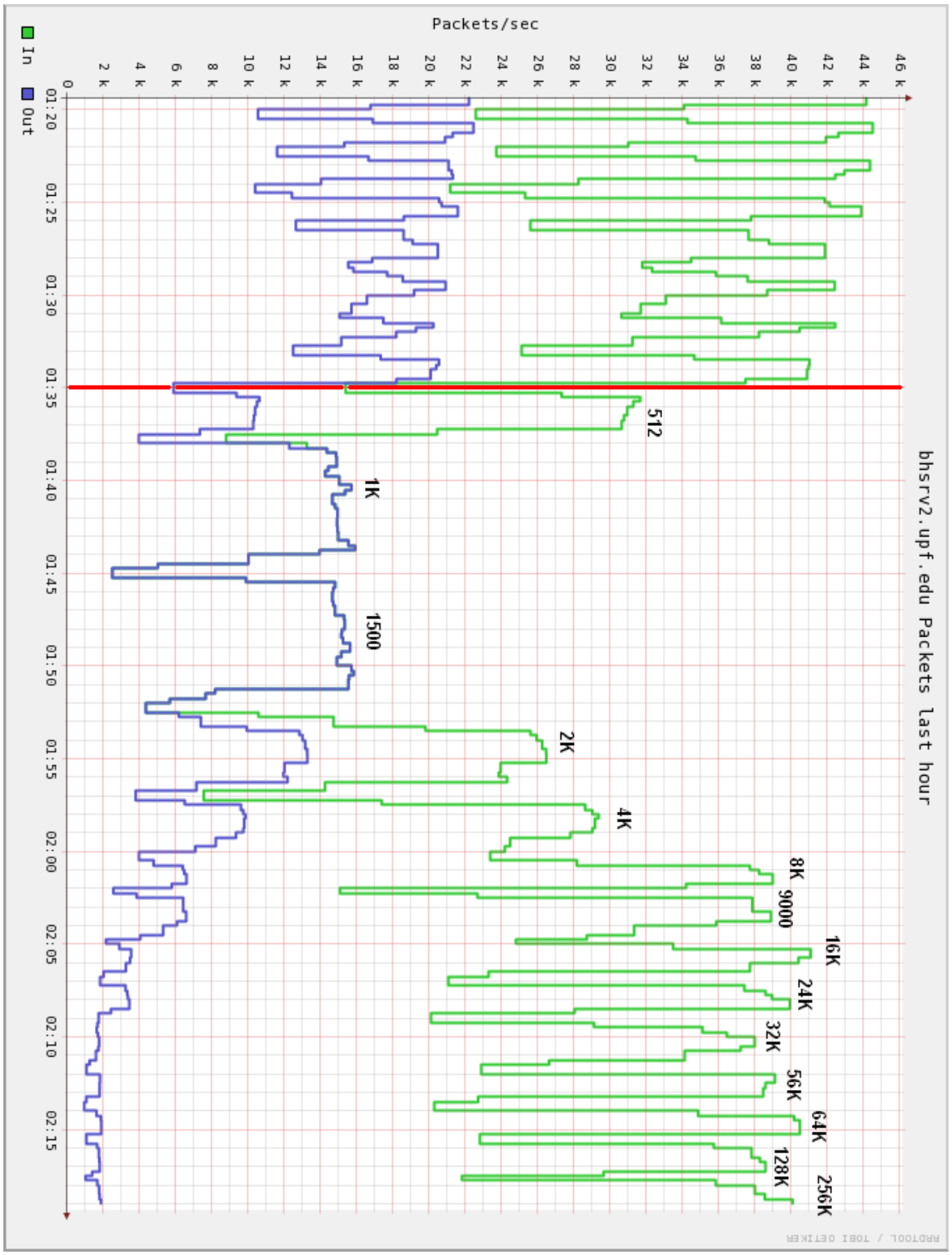

Figura 6.6 – Tasses de transmissió de paquets de la xarxa durant les proves de 4 escriptures simultànies amb 4 threads, nfs v3, UDP i mtu=1500.

#threads nfsd	#simult anis	opcions	protocol	mtu	RSIZE													
				BS=	512b	1K	1500	2K	4K	8K	9000	16K	24K	32K	56K	64K	128K	256K
	LOCAL				258,22	269,18		276,38	282,10	278,43		271,11	251,75	282,06	250,57	268,96	230,89	264,57
	LOCAL EN CACHÉ				682,15	1084,09		1499,08	1959,77	2219,34		2262,42	2231,29	2307,11	2367,85	473,99	2349,94	2381,39
4	1		tcp	1500	50,62	29,13	28,63	41,39	51,04	70,19	70,55	86,94	88,46	106,51	106,69	105,27	105,87	106,18
		cache			1802,14	1815,27	1810,26	1811,06	1819,29	1815,62	1816,59	1823,04	1816,62	1816,72	1819,36	1823,48	1822,40	1822,78
		cache 2			1901,03	1908,34	1905,51	1902,36	1904,11	1915,03	1914,00	1905,37	1906,20	1911,57	1911,92	1913,96	1901,96	1908,39
4	1		udp	1500	53,19	36,57	36,50	52,41	55,94	71,48	72,19	94,24	93,78	107,33	106,32	106,68	106,06	104,87
		cache			1814,84	1818,88	1819,23	1817,37	1821,31	1814,88	1821,80	1814,30	1815,30	1820,96	1821,77	1822,05	1819,53	1823,23
		cache 2			1906,95	1909,03	1909,18	1906,60	1899,52	1911,63	1906,93	1905,37	1913,92	1913,04	1915,08	1910,61	1912,36	1910,99
4	24	wdelay	tcp	1500	94,21	35,98	25,43	137,79	93,43	96,14	92,36	96,79	96,64	96,07	96,21	96,21	94,43	94,64
4	24	wdelay	udp	1500	101,74	45,00	45,83	69,14	93,79	95,57	96,29	105,23	105,46	105,69	105,23	105,62	107,08	106,69
64	24	wdelay	tcp	1500	94,2143	35,9783	25,4286	137,786	93,4286	96,1429	92,3571	96,7857	96,6429	96,0714	96,2143	96,2143	94,4286	94,6429
64	24	wdelay	udp	1500	101,739	45	45,8292	69,1429	93,7857	95,5714	96,2857	105,231	105,462	105,692	105,231	105,615	107,077	106,692

Taula 6.2 – Velocitat de lectura de disc en MB/s.

Conclusions i resultats

Amb les dades que hem obtingut (*taules 6.1 i 6.2*), arribem a diverses conclusions:

- Hem d'actualitzar el kernel. SuSE Linux Enterprise és una distribució que utilitza versions "antigues però provades". Això fa que determinades opcions que en el seu moment eren "noves", no funcionin correctament. Més concretament hem comprovat que resulta impossible servir volums NFSv4 degut a un error intern de idmap, i que la configuració de NFSv3 amb una MTU de 9000 es torna inestable sota càrrega, provocant que el sistema es bloquegi.

- Tot i que hem vist que passant de MTU 1500 a 9000 obtenim una lleugera millora de rendiment, degut al punt anterior en descartem el seu ús.

- TCP i UDP no presenten diferències notables en la velocitat de transmissió, però sí en el nombre de paquets, essent amb UDP una tercera part que amb TCP. De moment l'excés de paquets no ens desborda, però caldrà tenir-ho present en cas que ampliéssim "molt" el nombre de nodes de càlcul. Per altra banda, a l'ampliar el nombre de nodes augmentaran les col·lisions de paquets, empitjorant el rendiment de UDP respecte a TCP, així que no podem extreure'n una conclusió i caldrà que ho testem en cas que en un futur ens trobéssim en aquesta situació.

- Wdelay: Pràcticament no hem trobat diferències de rendiment amb l'aplicació o no d'aquest paràmetre. En els tests amb 1 sol procés escrivint, no apreciem cap diferència en cap de les combinacions de mida de bloc escrit i mida de la finestra d'escriptura. Trobem una lleugera diferència de velocitat de menys d'un 5% en les proves amb més processos escrivint que threads del servei (probablement degut a que els threads del servei queden "bloquejats" esperant a completar l'escriptura).

- Wsize: Els resultats indiquen que el rendiment de les escriptures arriben al seu màxim a l'establir la finestra d'escriptura a 32KB, el límit establert pel kernel del servidor nfs. Augmentar el valor per sobre de 32KB no afecta al rendiment ja que nfs el negocia entre client i servidor fixant-lo al mínim dels dos (32KB en

aquest cas).

- Rsize: Igual que en el cas de wsize, obtenim el màxim rendiment a l'establir la finestra al límit de 32K.

- Xarxa: En el cas de les lectures, la xarxa es torna un factor limitant. Les lectures de disc local es realitzen a 250 MB/s, mentre que la nostre xarxa, al tenir una capacitat de 1Gbps, té la seva velocitat limitada a un màxim teòric de 125MB/s. Degut a això, la xarxa es torna un coll d'ampolla a les lectures, donant-nos un rendiment final de 100~115MB/s. En el cas de les escriptures no, al ser la velocitat màxima d'escriptura local de ~65MB/s.

- Cache: Degut a la topologia de la xarxa que tenim, ens trobem amb dues cachés que poden afectar al rendiment: la del servidor i la del client. Pel que fa a les lectures, el paper de la cache del client és dramàtic, accelerant la velocitat de lectura dels ~100MB/s a ~1.9GB/s. En canvi, el paper de la cache del servidor és molt menor degut al coll d'ampolla que suposa la velocitat de la xarxa.

Així doncs, amb tota aquesta informació, optarem per a modificar la configuració del servei de NFS per a deixar-la de la següent manera:

Servidor:

```
Mtu: 1500
Threads: 64
wdelay: sí
gss: no
nfs4: no
```

Client:

```
Mtu: 1500
Protocol: tcp
Nfs3
Wsize: 32KB
Rsize: 32KB
```

Clúster de càlcul

Dins del clúster de càlcul, els serveis que haurem de comprovar són el rendiment de Sun Grid Engine i el funcionament de les llibreries de paral·lelització.

SGE

Al finalitzar la instal·lació dels nodes, el sistema gestor de cues SGE ja és plenament funcional, tot i que caldrà modificar la configuració per a millorar el sistema. Inicialment desconeixíem quina configuració s'adaptaria millor a les necessitats dels jobs dels usuaris, i vam optar per posar el sistema en producció amb la configuració bàsica. Amb el pas del temps el vam anar adaptant en diverses iteracions a mesura que l'ús del sistema i el nostre coneixement sobre ell anaven evolucionant.

Primera iteració

La primera iteració del sistema consistia en utilitzar la configuració inicial del SGE. Aquesta configuració es pot resumir en l'ús d'una única cua “all.q” que conté a tots els nodes de càlcul. Cada node pot admetre fins a 8 jobs (un per core), i els usuaris es reparteixen els jobs utilitzant una cua FIFO.

Valoració:

El principal inconvenient que presentava aquest sistema era la cua FIFO i el jobs llargs llençats en batch (diversos processos similars amb arxius d’entrada diferents enviats simultàniament). Si un usuari decidia enviar un gran nombre de treballs a la cua, en distribuir-se els jobs utilitzant una cua FIFO, quan entrava el primer job de l'usuari, els següents treballs que s’executarien al sistema eren també de l'usuari, fins acabar tots els treballs del batch que hi havia en espera. Si aquests jobs tenien una duració una mica llarga (2-3 dies o més), això resultava en un únic usuari ocupant els 64 slots de càlcul dels que disposa el sistema durant un temps indeterminat, impedint a la resta d'usuaris executar cap mena de procés.

Segona iteració

Després d'analitzar els problemes descrits a la primera iteració, els vam intentar resoldre eliminant la cua única "all.q" i creant-hi quatre cues:

- *Slow*: cua amb accés exclusiu a 4 nodes de càlcul (32 slots). Sense límit de temps
- *Medium*: cua amb accés exclusiu a 3 nodes de càlcul (24 slots). Límit de temps de 24 hores. En aquesta cua els jobs s'executen amb un límit de 24h de temps real d'execució. Si el límit es supera, es cancel·la automàticament el job (avisant a l'usuari via mail).
- *Fast*: cua amb accés exclusiu a1 node de càlcul (8 slots). Límit de temps d'1 hora. Els jobs que s'executen durant més d'una hora es cancel·len.

L'objectiu d'aquest sistema era repartir l'ús que es feia del clúster entre els diversos usuaris segons el tipus de tasca que haguessin d'executar, de tal manera que un sol tipus de job (especialment els llargs) no saturessin el sistema deixant-lo inservible per la resta d'usuaris. Podria saturar-se per un dels 3 tipus de job, però no pels altres.

Vam crear també una nova cua per a proves:

- *test_queue_5_min*: cua amb accés a 1 slot a cadascun dels nodes de càlcul. Els jobs es cancel·len passats els 5 minuts d'execució.

L'objectiu d'aquesta cua era permetre als usuaris executar breument els seus processos al clúster per comprovar que estan ben adaptats a l'entorn i no haver d'esperar així tota la cua per a descobrir que el seu job no funciona perquè, per exemple, han escrit malament el path d'un fitxer. Aquesta cua es va distribuir entre tots els nodes per igual. D'aquesta manera es podrien executar als nodes amb menys càrrega o, en el pitjor dels casos, elevar la càrrega dels nodes ocupats de 8 a 9 processos (molt menys traumàtic que elevar la càrrega d'un sol node a 16).

L'ús correcte d'aquestes cues quedava sota responsabilitat dels propis usuaris, els quals van ser convenientment informats del canvi realitzat al sistema per a que especulessin sobre el temps d'execució dels seus programes.

Addicionalment a aquest canvi a les cues, vam modificar el sistema d'espera per a passar del sistema FIFO a un sistema més just. La configuració de la política del SGE s'estableix en 3 elements:

- Prioritat:
  La prioritat donada a un job. Per defecte tots els jobs tenen la mateixa prioritat, però l'usuari pot rebaixar la prioritat de qualsevol dels seus jobs per a que altres s'executin abans. Només l'administrador pot augmentar la prioritat d'un job.

- Política d'urgència:
  Urgència que té un determinat job. S'estableix assignant el "pes" que tindran dos paràmetres:
  - *Deadline jobs*: A l'enviar un job, els usuaris que tinguin permís per a executar jobs amb deadline (hauran de ser inserits manualment en el grup "deadline" per l'administrador), podran donar una data límit per a que aquest comenci a executar-se. La prioritat del job s'anirà incrementant a mesura que s'apropi la data límit, arribant a la prioritat màxima en arribar a aquesta data.

  - Temps d'espera: Temps que porta el job esperant a la cua per a executar-se.

- Política de tiquets:
  Política que determinarà el repartiment de tiquets entre els diferents usuaris, departaments, projectes o jobs. Aquesta política es divideix en 3 tipus de tiquets:

  - *Share Tree*: Política de l'arbre de recursos compartits. Permet crear un arbre el qual distribueixi mitjançant percentatges, el "dret a usar recursos" que tenen els diferents projectes (i els diferents usuaris dins d'un mateix projecte). D'aquesta manera ens permetrà configurar polítiques com per exemple, donar un 70% de recursos als jobs d'un projecte determinat, i que el 30% restant es reparteixi entre la resta d'usuaris. També permet configurar com es calcularà aquest "ús de

recursos" especificant el pes relatiu que li donem als diferents recursos "consumibles" del sistema: temps de cpu, memòria i entrada/sortida.

- *Tiquets funcionals*: Similar als tiquets del "*share tree*", però enlloc de distribuir segons l'ús dels recursos, distribueix per nombre de jobs. Permet configurar la distribució de tiquets entre projectes i usuaris, i també entre departaments o jobs, encara que de manera menys complexa que amb "share tree".

- *Override tickets*: Permet assignar a determinats usuaris, projectes o departaments la capacitat d'obtenir, a curt termini, més recursos dels que li pertocarien.

Sabent tot això, veiem que el comportament FIFO que estàvem tenint fins al moment era degut a que la configuració per defecte de SGE no tenia en compte la política de tiquets. Per a resoldre això i tenir una política més justa per als usuaris, vam configurar els pesos i els tiquets de la següent manera:

- Pesos:
  - Prioritat: 1
  - Urgència: 0.1
    - Deadline: 3.6e+06 (valor per defecte)
    - Temps d'espera: 0
  - Tiquets: 0.01
- Tiquets:
  - share tree: 100000000 (nombre arbitràriament gran, anteriorment 0)
  - functional: 100000000 (nombre arbitràriament gran, anteriorment 0)
  - override: 0

Tal i com passava anteriorment, aquesta configuració donava el màxim pes a la prioritat del job, però aquesta passaria a ser negligible degut al fet que tots els jobs tenen la mateixa per defecte. El pes de la urgència del job el vam deixar com estava, donant màxim pes a la data de deadline (en cas que calgués fer-la servir), seguida del temps d'espera que, per norma general, no tindria cap efecte degut a que el propi sistema dóna per defecte més prioritat als jobs més antics. D'aquesta manera, si no hi intervingués

cap altra causa, els jobs que enviés un mateix usuari s'executarien sempre en ordre. Finalment vam donar un pes baix als tiquets, tot i que seria el més important de tots ja que la resta de valors en la majoria de casos no variarien.

Pel que fa als tiquets, vam establir-los a un nombre arbitràriament alt, tal i com recomana la documentació. Vam donar el mateix pes tant als tiquets funcionals com als de "*share tree*", per tal que tant el repartiment "igualitari" com la regulació de l'ús de recursos tinguessin un cert pes en la distribució dels jobs. En aquest punt podríem haver afegit opcions molt potents de distribució del percentatge d'ús dels recursos que podrien fer els usuaris que pertanyessin al grup dels diferents investigadors principals, però, tot i plantejar aquesta alternativa, els caps del departament van decidir no utilitzar-la en aquell moment. El mateix va succeir amb els tiquets d'override per a permetre a determinats usuaris passar per davant d'altres.

Valoració:

Després d'un temps en producció, aquesta política va demostrar ser eficient, especialment la configuració dels tiquets més que en allò referent a les cues. El sistema funcionava perfectament en cas de trobar-nos en un escenari on s'executés la mateixa quantitat de tasques de cada tipus (o hi hagués un ús habitual de tots tres tipus de jobs, lents, mitjans i ràpids), però això només era així puntualment. En la majoria de casos ens trobàvem amb un ús massiu de només un dels tres tipus de job. Això provocava problemes com embussos continus a la cua "slow" quan algun usuari llençava diversos jobs que podien arribar a trigar més d'un mes, mentre els nodes assignats a cues "medium" i "fast" estaven totalment infrautilitzats.

Tercera iteració

Per a suplir els problemes detectats durant la segona iteració vam modificar els paràmetres de temps de les cues per tal d'augmentar el marge de cadascuna. D'aquesta manera, el temps límit de la cua "*fast*" va passar d'1 a 24 hores, i el de la cua "*medium*" de 24 hores a 3 dies. Addicionalment, vam modificar la distribució dels hosts, passant un dels que estaven assignats a la cua "*medium*" cap a la "*slow*", quedant amb una distribució 5-2-1.

Tot i els canvis realitzats en les anteriors configuracions, vam notar que en determinades ocasions s'acumulaven, en un mateix node, un conjunt de tasques amb requisits de memòria molt grans. Sovint els requisits no eren "reals", sinó deguts a un mal dimensionament de la memòria o, sobre tot, a l'ús de conjunts de dades massa grans en programes que no estaven pensats per a suportar-los. Això provocava que els processos ocupessin tota la memòria física del node i fins i tot tota la swap, deixant el node bloquejat de manera temporal o indefinidament fins a rebre intervenció humana.

Per tal d'evitar aquests col·lapses, vam decidim utilitzar els "atributs consumibles" que ofereix SGE: paràmetres dels quals se n'assigna una determinada quantitat a un host o cua, i que els jobs que hi entrin poden reservar per a ells mentre siguin disponibles. D'aquesta manera, vam declarar com a consumible l'atribut "`h_vmem`", el qual es corresponia amb el límit "dur" de memòria virtual que pot utilitzar un job (està relacionat amb la comanda de sistema "*ulimit*"). Vam assignar 17GB d'aquest atribut a cada node, de manera que fos proper a la suma de la memòria física i la swap (deixant 1GB lliure per al sistema en cas de "crisi"). També vam definir 2GB com a valor per defecte (el valor que s'utilitzaria en cas que l'usuari no demanés explícitament l'atribut). Aquest valor ens permetia "encaixar" 8 jobs en un mateix host i, segons les estadístiques d'ús del sistema, fins a aquell moment més d'un 80% dels jobs utilitzaven menys de 2GB.

Valoració:

Definides aquestes modificacions, vam comprovar que el funcionament del sistema millorava força, especialment pel que fa a les caigudes per excés de "swapping". Tot i això, seguíem tenint un problema amb les cues. Mentre les cues estiguessin vinculades "fortament" a un tipus de job, l'excés o manca de jobs d'un determinat tipus provocaria un coll d'ampolla en una de les cues o una infrautilització de la resta de nodes.

Per altra banda, tot i que el canvi en l'apartat de memòria va millorar el funcionament del sistema, també provocava problemes: El sistema de cues de SGE no "espera" els processos, sinó que intenta ocupar el màxim nombre de slots de cpu (*figura 6.7*). Si un procés demana més memòria de la que pot oferir cap màquina, i la resta de processos en cua en demanen menys, fins que no es creï un "buit" suficientment gran a un node, el procés no podrà executar-se. Degut a això, en un ambient on hi hagués abundància de

treballs amb pocs requisits de memòria, i uns pocs jobs amb grans requisits, els jobs petits inundarien el sistema "colant-se" sempre davant dels jobs grans.

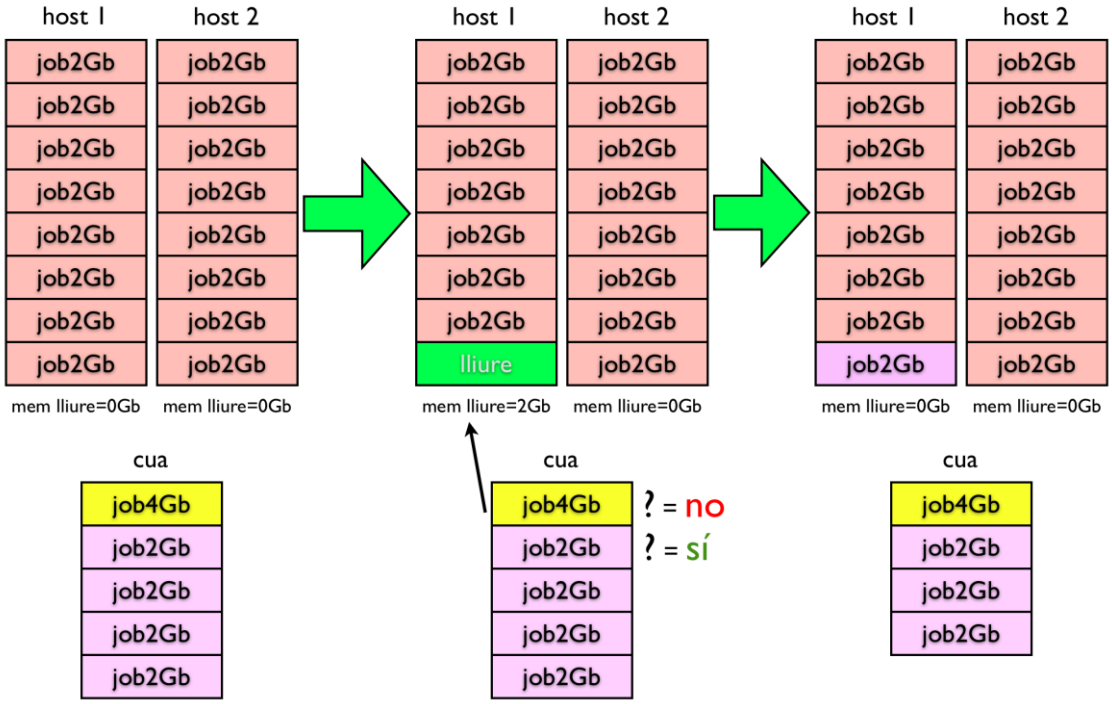

Figura 6.7 – Exemple de bloqueig d'un procés degut a l'ocupació de memòria.

Quarta iteració

Aprofitant una ampliació de memòria dels nodes de càlcul, els quals van passar de tenir 9GB a tenir-ne 32, 16 o 8 (4 nodes amb 32GB, 3 amb 16GB i un amb 8GB), vam modificar també el sistema de cues per a solucionar els problemes anteriors i adaptar-nos al nou escenari. Gràcies a l'increment de memòria disponible, vam decidir evitar l'ús de la swap en la mesura que fos possible. Per això vam rebaixar els límits de memòria disponibles a cada node fins a la memòria física disponible realment a cada node.

Degut a la nova distribució de memòria variable entre nodes, vam optar per una nova estratègia de cara a la distribució dels jobs. Enlloc d'intentar separar-los per la seva durada, vam decantar-nos per intentar omplir el major nombre possible de slots. Per tal d'assolir aquest "ús ideal" del sistema, vam modificar la configuració de cues de manera que ens permetessin afavorir que els jobs s'executessin al seu node "ideal". És a dir: els jobs que demanessin 1GB o menys de memòria intentessin entrar preferentment al node

de 8GB (ja que 8 jobs * 1GB = 8GB). Els que demanessin entre 1 i 2GB anessin als nodes de 16GB. I els jobs de més de 2GB als nodes de 32GB.

Per a fer això vam separar els hosts en 3 grups, segons la seva memòria. Vam crear també 3 cues, *all-8*, *all-16* i *all-32* les quals prioritzarien l'accés dels jobs als nodes corresponents. Això, però, no seria forma exclusiva, ja que no volíem repetir l'error de tenir la meitat del sistema buit al no haver-hi en cua jobs d'un determinat tipus. Per a evitar-ho, va caldre configurar la cua amb "nombres de seqüència" per tal que es consultés la disponibilitat de slots als diferents nodes en ordre, primer el més adequat i després la resta.

Finalment vam crear un script "embolcall" que substituiria el mecanisme d'enviament de jobs habitual, el qual decidiria a quina cua s'hauria d'executar cada job segons la memòria que hagués demanat l'usuari. D'aquesta manera l'elecció de cua seria transparent per a l'usuari.

Preveient que aquest sistema provocaria problemes amb els usuaris que no fessin un ús correcte de la RAM (sobre-requeriment de memòria que impedís l'entrada d'altres tasques), vam crear també un sistema de penalització. Aquest es va plasmar en un script que diàriament revisaria l'ús de memòria dels jobs de cada usuari, i els penalitzaria en cas de reservar un excés de memòria, reduint el nombre de tiquets de l'usuari infractor de tal manera que la prioritat dels seus jobs es veiés reduïda.

Valoració:

Amb aquesta darrera iteració, el funcionament general del sistema de cues va millorat moltíssim. Si bé és cert que el sistema no era perfecte i que hi havia saturacions en determinats moments puntuals, l'ús general era molt fluid i presentava un gran aprofitament dels recursos. Si bé aquesta nova política depenia molt del bon ús que en fessin els usuaris, els caps del departament van acordar fomentar-ne l'ús adequat, educant-los en el seu funcionament i de manera que els possibles problemes que sorgissin en aquest àmbit es resolguessin "fora del sistema".

Cinquena iteració

La configuració anterior presentava tres problemes principals:

- Mal ús de memòria per part dels usuaris:
  Per a solucionar-ho vam creat un script el qual permetés als usuaris veure l'estat actual del sistema: nombre de nodes, jobs i memòria reservada i usada per cada procés. D'aquesta manera els propis usuaris podien veure si algú estava fent-ne un mal ús. En cas d'usuaris reincidents aquests problemes serien resolts "fora del sistema".

- Saturació de tot el sistema per un únic usuari:
  Amb l'última iteració va tornar a ser possible, com als inicis del sistema, que un únic usuari ocupés tots els nodes de càlcul. Per a solucionar-ho vam configurar la restricció del màxim nombre de jobs per usuari a 50. L'ús d'aquesta política ja l'havíem plantejat en un inici, però havia estat descartada pels caps per motius estratègics.

Valoració:
Amb aquesta última iteració, el funcionament general del sistema és molt bo, pràcticament perfecte. L'únic però que se li podria posar és un cert desaprofitament de la memòria en certs moments degut a que els usuaris n'han de demanar una mica més de la que necessitaran (un petit marge per a imprevistos o simple arrodoniment al valor més proper). Això també serà provocat per processos que tinguin un "pic" de memòria en un moment determinat i que hagin de reservar la memòria tenint en compte aquest pic tot i que durant la resta de l'execució no la necessitin. Salvant això, sempre quedaran els conflictes puntuals entre els usuaris o dates límit, però aquests temes s'hauran de tractar individualment i des d'un punt de vista polític aliè al funcionament del sistema.

OpenMP / MPI

Durant la instal·lació de Rocks-cluster vam instal·lar també el paquet HPC-rocks (High Performance Computing) el qual inclou la instal·lació de MPICH i OpenMPI per a

permetre la paral·lelització dels programes que estiguin preparats per a ells. Des de la instal·lació del sistema han sortit noves versions d'aquests llenguatges que inclouen novetats imprescindibles per a alguns dels programes que calen als usuaris. Hem hagut d'instal·lar aquestes noves versions pel nostre compte, fora de l'entorn proporcionat pel paquet HPC-rocks, de manera que hem hagut de testar-ne el funcionament.

Inicialment, ens trobem que els programes no funcionen correctament degut a que, quan el procés "master" crea els processos esclaus, la identificació de l'usuari es realitza mitjançant ssh, i tots els nodes de càlcul demanen el password de l'usuari per a poder accedir. Per tal de resoldre això, només caldrà afegir a la configuració la opció que indica que la xarxa connectada a la interfície eth0 és "segura" i amb això els processos funcionaran correctament.

Funcionament i rendiment:

Comprovem que la paral·lelització funciona adequadament, i que no es produeixen retards que interrompin la comunicació master-esclau. Per exemple, un procés paral·lelitzat en 8 processos triga 140,3 segons, mentre el mateix procés dividit en 53 triga 41,8 segons, de manera que, tot i que la velocitat del procés no creix linealment (cosa que tampoc esperàvem), sí que notem una millora de la velocitat molt important.

Integració amb SGE:

Per tal d'executar processos paral·lels dins de l'entorn del gestor de cues, SGE inclou entorns de paral·lelització que adapten els paràmetres i recursos de SGE als paràmetres i opcions que necessiten MPICH i OpenMPI per a paral·lelitzar els seus processos (per exemple, el nombre de slots lliures), així com scripts per a la inicialització de la paral·lelització. Després de testar-ho, comprovem que els entorns de paral·lelització "antics" de SGE funcionen a la perfecció amb les noves instal·lacions.

Estadístiques d'ús

A continuació mostrem algunes de les gràfiques sobre l'ús que s'ha fet del clúster des de l'1 de gener de 2009 fins a l'1 de novembre de 2010. Dividirem aquestes gràfiques en 3 categories:

- Ús del sistema.
- Profiling dels jobs.
- Ús dels usuaris i funcionament de la política de penalització.

Ús del sistema

Avaluarem l'ús que s'ha fet del sistema analitzant l'aprofitament que s'ha fet dels seus recursos durant aquest període. Això ho podem mesurar en l'ús de CPU, de memòria i de tots dos recursos en conjunt.

Mesurarem la ocupació mensual del clúster com la suma del temps d'execució (en segons) de tots els processos que s'hi han executat, dividida pel total de temps que ha ofert el clúster (8 nodes x 8 processos x 30 dies x 24 hores x 3600 segons). Aquestes estadístiques estan lleugerament esbiaixades, ja que només es té en compte la data d'inici del procés. Si un job s'executa durant diversos mesos, es sumarà tot al primer mes, encara que comenci el dia 31 d'un més i acabi el dia 25 del següent. Per a compensar això, calculem també tots els estadístics, però separant-lo per trimestres per a reduir-ne l'efecte.

Les estadístiques d'ocupació mensual (*figura 6.8*) i trimestral (*figura 6.9*) del clúster, ens mostren que s'ha fet un ús irregular dels recursos de CPU. Podem observar que s'alternen moments de poca ocupació, amb pics de molt treball. Això és degut a la natura de la recerca que s'hi realitza així com també a l'efecte dels batchs de processos. Tot i això, l'estadístic R^2 ens mostra una tendència d'ús clarament creixent.

Pel a mesurar l'ús de memòria no podem fer servir la memòria real utilitzada per cada procés ja que aquesta no resultarà prou informativa. Enlloc d'això calcularem l'àrea mínima de la memòria que hauria hagut de reservar el job en cas de trobar-se present la darrera política de cues. Per a fer això hem multiplicat el pic de màxima memòria pel temps d'execució de cada procés, i hem calculat l'ocupació mensual i trimestral de tot el sistema.

De cara a valorar aquestes estadístiques, però, hem de tenir en compte dues dates:

- A principis d'octubre de 2009 vam fer una parada per a instal·lar més memòria al clúster de càlcul, passant de 72 a 184GB en total.
- A principis d'abril de 2010 vam posar en producció el canvi en la política de cues "restringides" per memòria i la penalització d'usuaris que en fessin un mal ús.

Amb les dades davant (*figures 6.10 i 6.11*), veiem clarament que memòria, per la seva pròpia natura depenent de les necessitats dels diferents usuaris al llarg del temps, és molt poc previsible. Tot i això, podem veure de fons una tendència de creixement, però molt dispers, donant un patró de pics d'alta i baixa demanda més que d'un ús constant. Podem veure també en aquestes gràfiques l'efecte de l'ampliació de memòria. Anteriorment ens trobàvem amb situacions com el juny de 2009, on, degut a la necessitat d'obtenir uns resultats abans de la lectura d'una tesi doctoral, ens vam veure obligats a solapar l'execució de diversos jobs, resultant en un ús de memòria superior a la que realment disposàvem (111,61%). Un cop realitzada l'ampliació, veiem que l'ús de memòria torna a ser més normalitzat, tot i que amb els seu patró propi de pics i baixades segons les necessitats del moment.

Analitzant els canvis produïts per l'increment de memòria del sistema, veiem que, abans del canvi teníem situacions (juny 2009) on, degut a la necessitat d'obtenir uns resultats de cara a la lectura d'una tesi doctoral, ens veiem tan restringits per la quantitat de memòria que vam haver de solapar diversos jobs donant com a resultat un ús de memòria superior a la que disposàvem (111,61%). També veiem que, després del canvi la tendència d'ús decreix lleugerament, però això és artefacte d'uns primers mesos amb una gran demanda, ja que durant el període de mínim ús després de l'ampliació, la memòria utilitzada està per sobre de la mitja que abans de l'ampliació.

Addicionalment, si comparem l'ús de memòria amb l'ús de cpu (*figures 6.12 i 6.13*) podem observar que hi ha una gran diversitat de tipus de jobs. Per norma general, els grans pics de memòria són deguts a un nombre limitat de processos amb grans necessitats, mentre que els períodes amb un menor ús de memòria solen ser deguts a un gran nombre de processos amb poques necessitats de memòria.

Finalment, pel que fa al canvi del sistema de cues d'abril de 2010, el qual canvia el paradigma i enlloc d'intentar fer el màxim ús de la CPU intenta maximitzar la memòria, en cas que el seu funcionament no fos correcte seria d'esperar una disminució de l'ús de CPU respecte al període anterior, però, per contra, observem un ús força saludable del sistema.

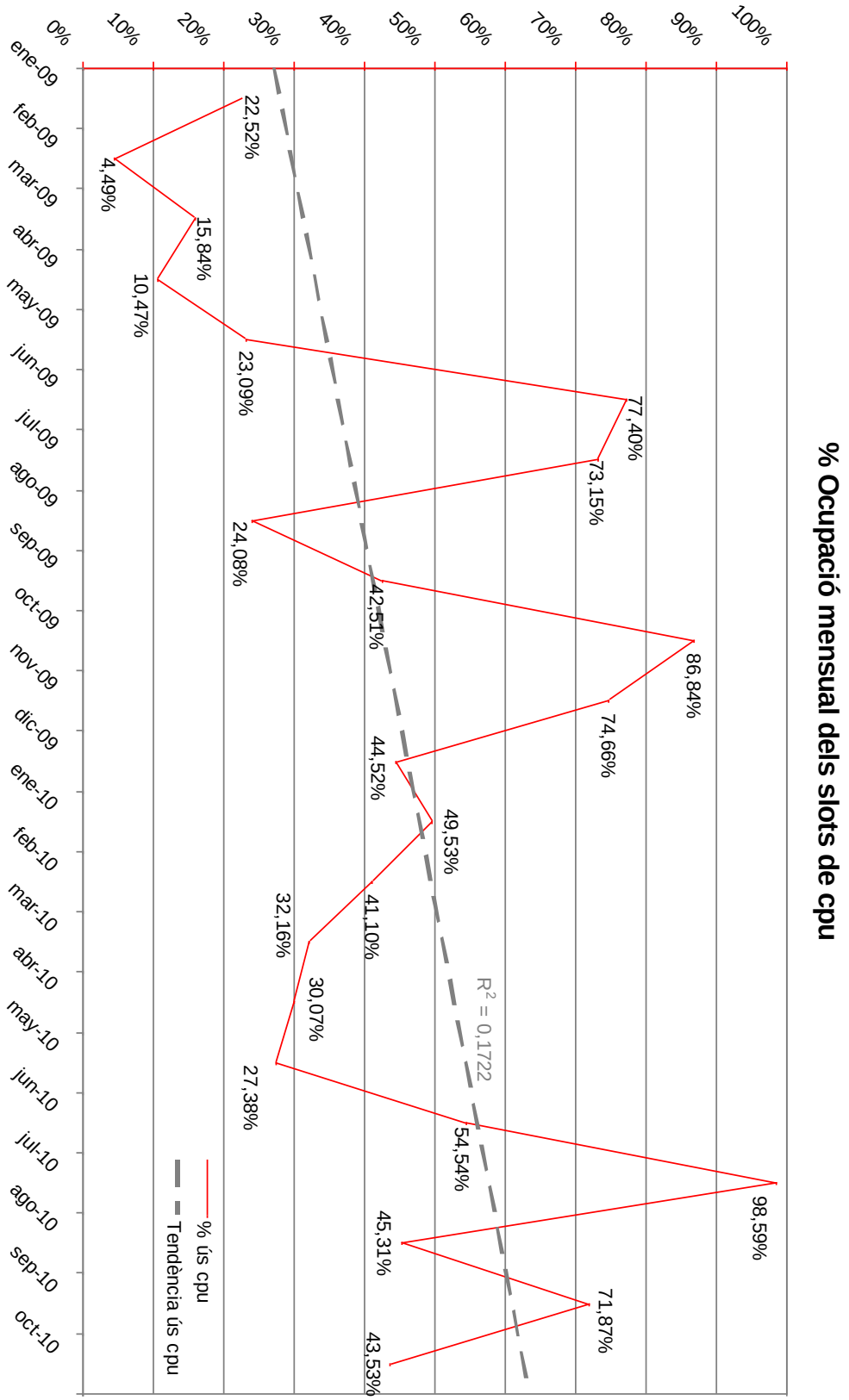

Figura 6.8 – Ús mensual de cpu, en % respecte al total ofert.

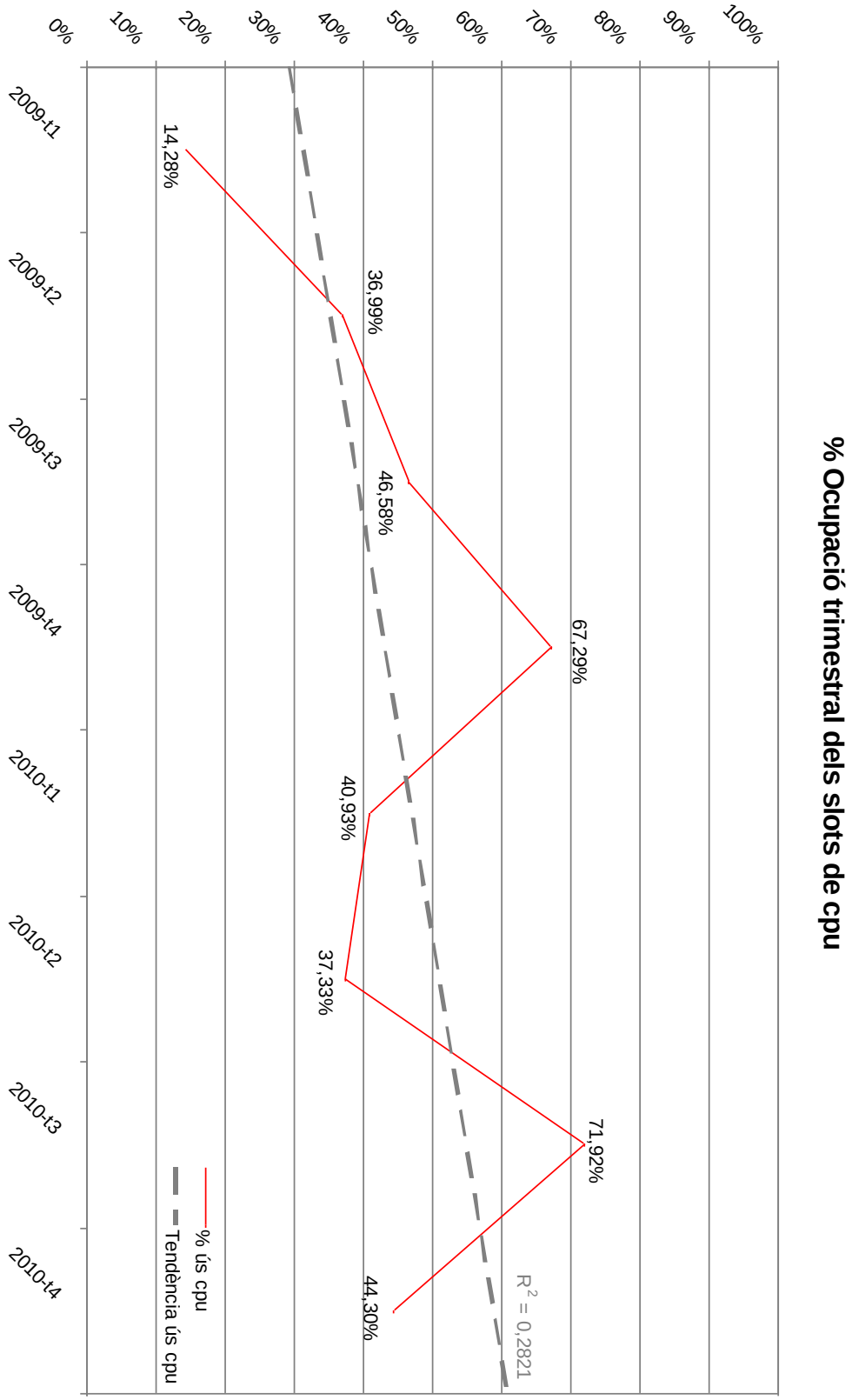

Figura 6.9 – Ús trimestral de cpu, en % respecte al total ofert.

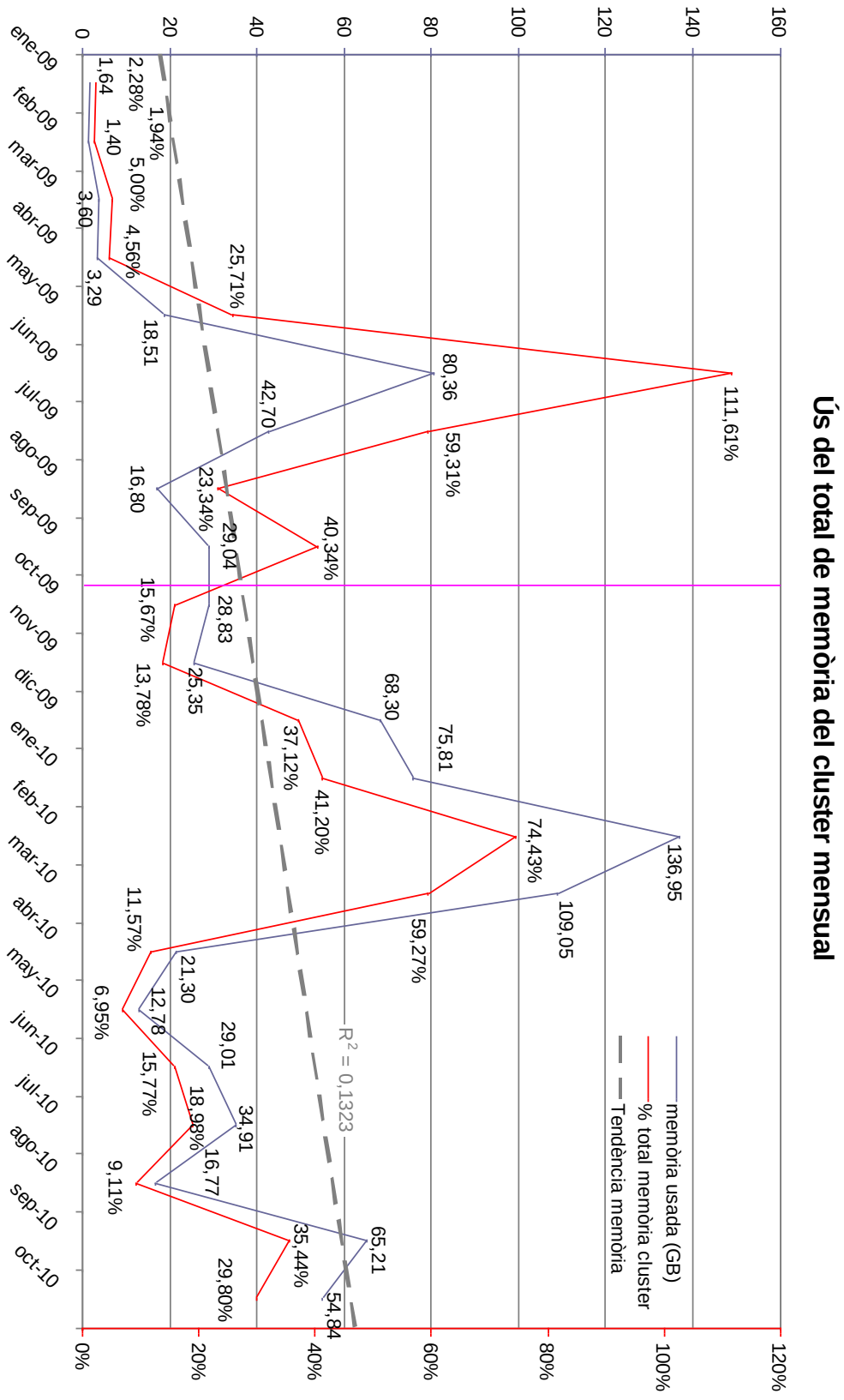

Figura 6.10 – Ús mensual de memòria, en % i memòria real. La línia indica la data en que vam realitzar una ampliació de memòria, passant de 72GB a un total de 184GB disponibles.

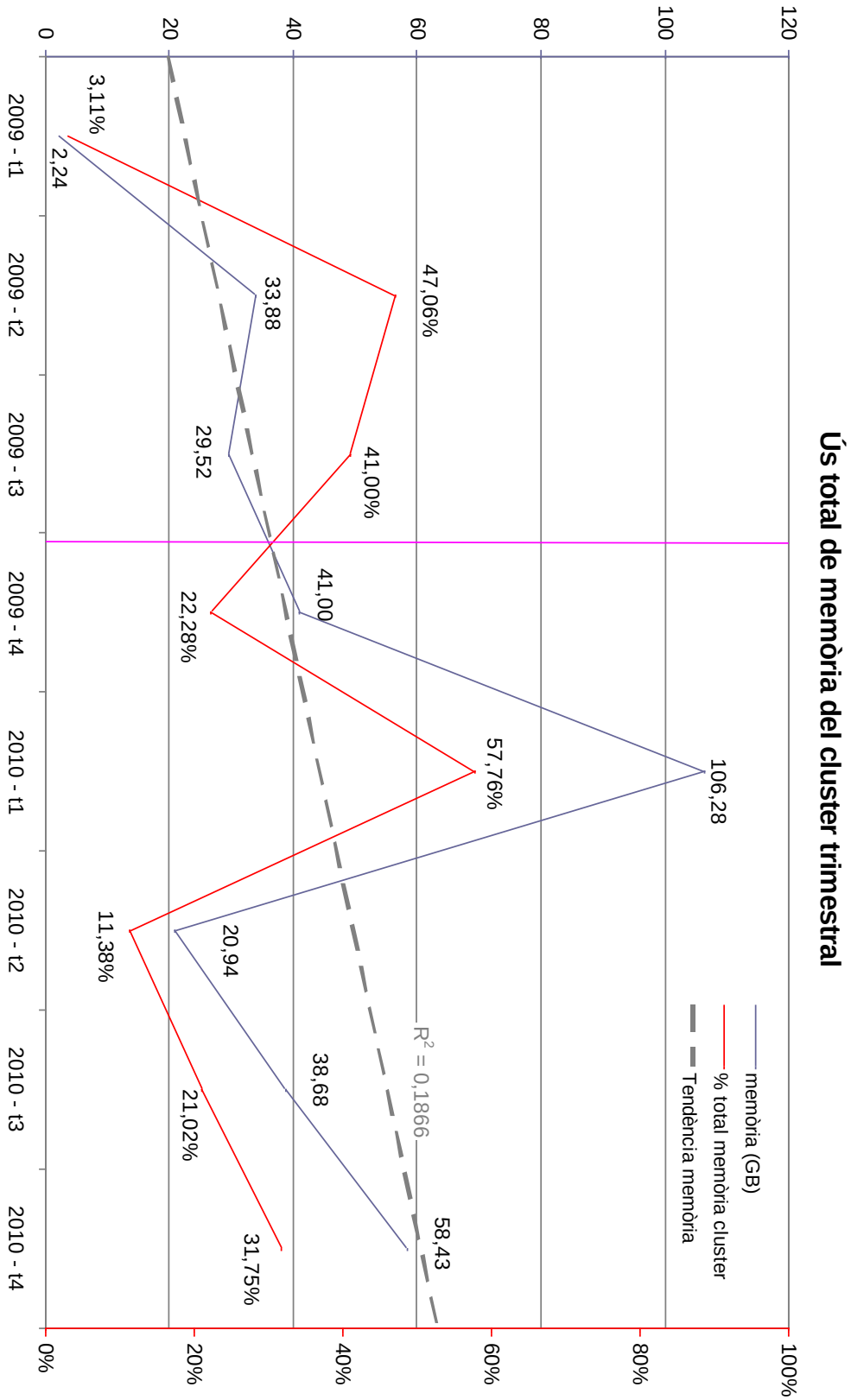

Figura 6.11 – Ús trimestral de memòria, en % i memòria real. La línia indica la data en que vam realitzar una ampliació de memòria, passant de 72GB a un total de 184GB disponibles.

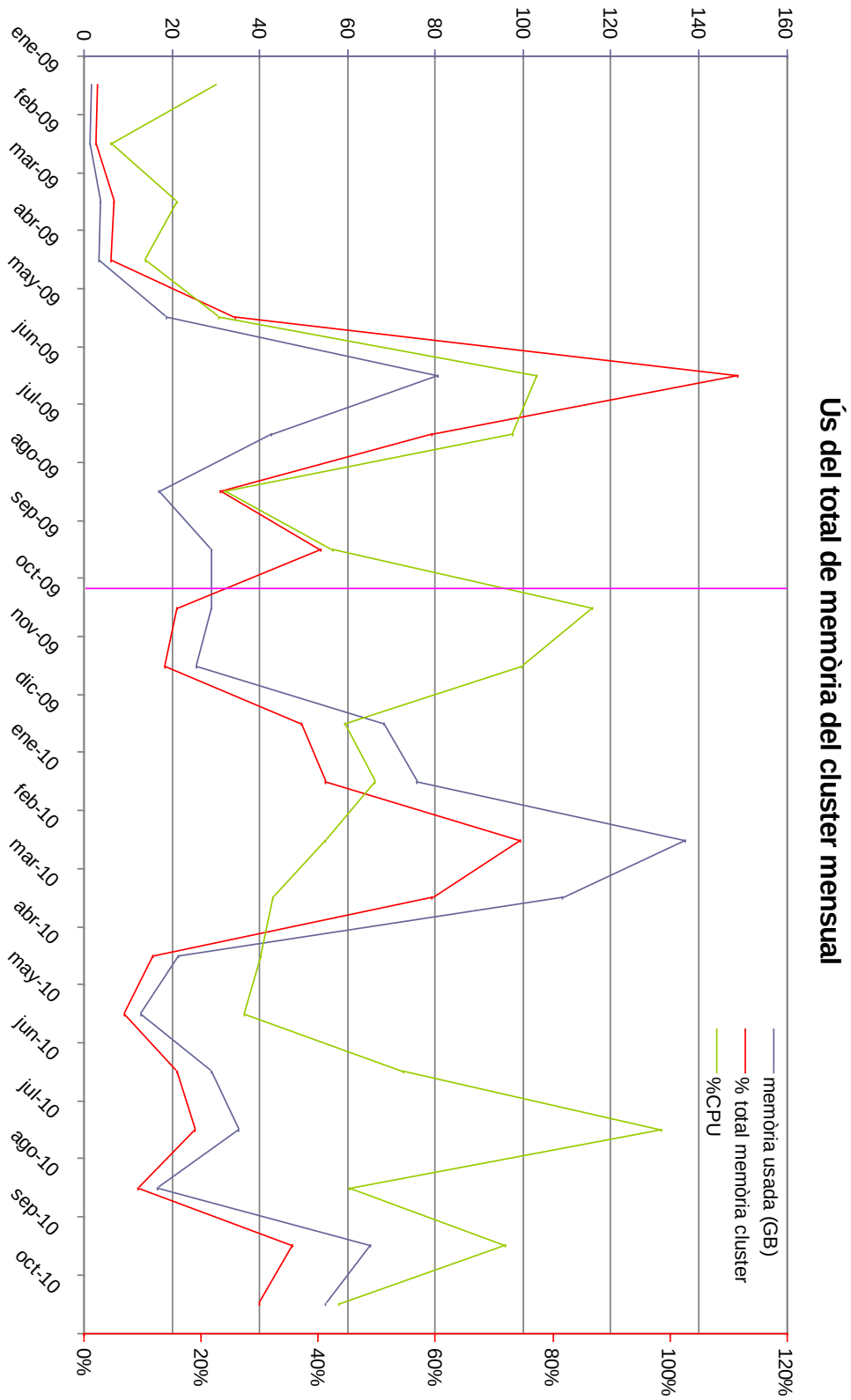

Figura 6.12 – Ús mensual de memòria, en % i memòria real, comparat respecte a l'ús de cpu.

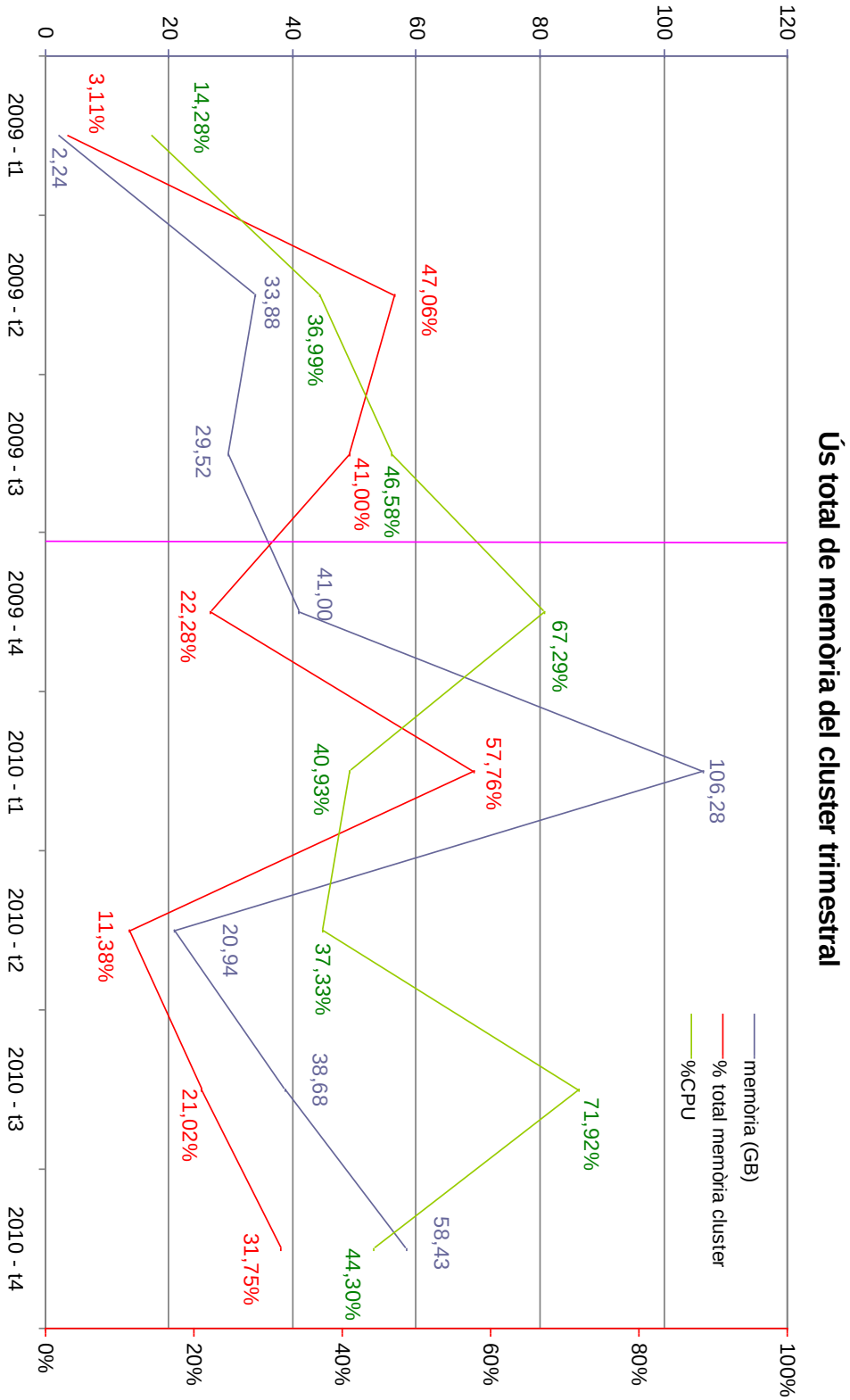

Figura 6.13 – Ús trimestral de memòria, en % i memòria real, comparat respecte a l'ús de cpu.

Profiling dels jobs

Analitzant els diferents jobs que s'han executat durant aquest període observem (*figura 6.14*) que tenim un enorme nombre de jobs que acaben quasi instantàniament. Aquesta mesura, però, és un artefacte de l'ús del sistema degut a que s'hi acumulen jobs realment ràpids, juntament amb tots aquells processos que els usuaris han executat malament (sigui dins del SGE o del propi programa). Realment no tenen cap impacte al sistema, ja que la suma del seu temps d'execució és de 1,12 dies, a penes un 0,0071% de l'ús total del sistema. Veiem també, en canvi, que els processos que més temps d'execució consumeixen es troben concentrats en la franja d'entre 12 hores i 2 setmanes, essent especialment destacables els de la franja d'entre 2 i 7 dies.

Pel que fa al temps d'espera en cua (*figures 6.15, 6.16 i 6.17*), veiem que un 60% dels jobs triga menys de 30 minuts en començar a executar-se, un 80% triga menys de 6 hores, un 90% triga menys d'un dia i un 99% triga menys de 2 setmanes. Això ens mostra que, tot i que de vegades el sistema es troba saturat, per norma general funciona de manera fluida. A la *figura 6.17*, el pic en el temps d'execució que trobem a la zona dels 2 a 7 dies, podria ser degut a la tendència d'enviar multitud de processos similars d'un mateix usuari. Això provoca que s'omplin tots els slots disponibles i els següents jobs en executar-se hauran d'esperar durant tota l'execució dels jobs, donant com a resultat un alt nombre de jobs "llargs" que han hagut d'esperar molt temps a la cua.

Pel que fa a la memòria necessària (*figura 6.18*), trobem que, tot i que tenim un gran nombre de processos que necessiten menys de 100MB (els quals possiblement inclouen diversos jobs erronis), el gruix dels nostres processos es troba a la franja d'entre 1 i 12GB, destacant especialment els que necessiten entre 4 i 6GB. En canvi, pel que respecta a l'ús que es fa de la memòria (*figura 6.19*), on entenem "ús" com el percentatge de l'àrea del màxim de memòria pel temps d'execució que realment s'ha utilitzat, trobem que la majoria de processos utilitzen entre un 40 i un 70% de la memòria que reserven. Això indica que no fan un ús constant de la memòria sinó que tenen alguns pics més o menys alts durant la seva execució. Tot i això, també tenim un bon nombre de processos pràcticament constants i, com podem veure (*figura 6.20*), el principal ús del sistema és per part d'aquests últims.

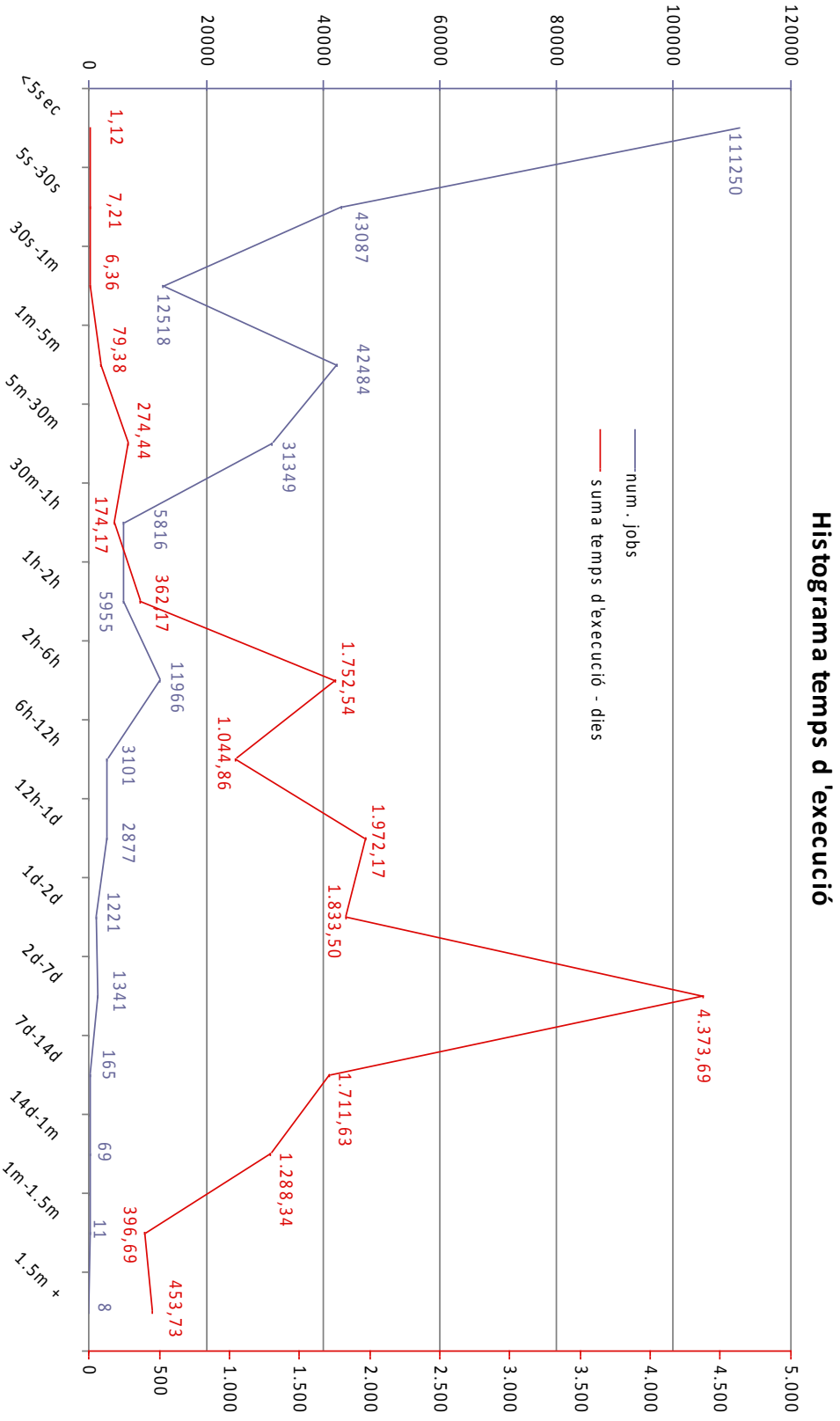

Figura 6.14 – Histograma del temps d'execució, en nombre de jobs i suma de dies.

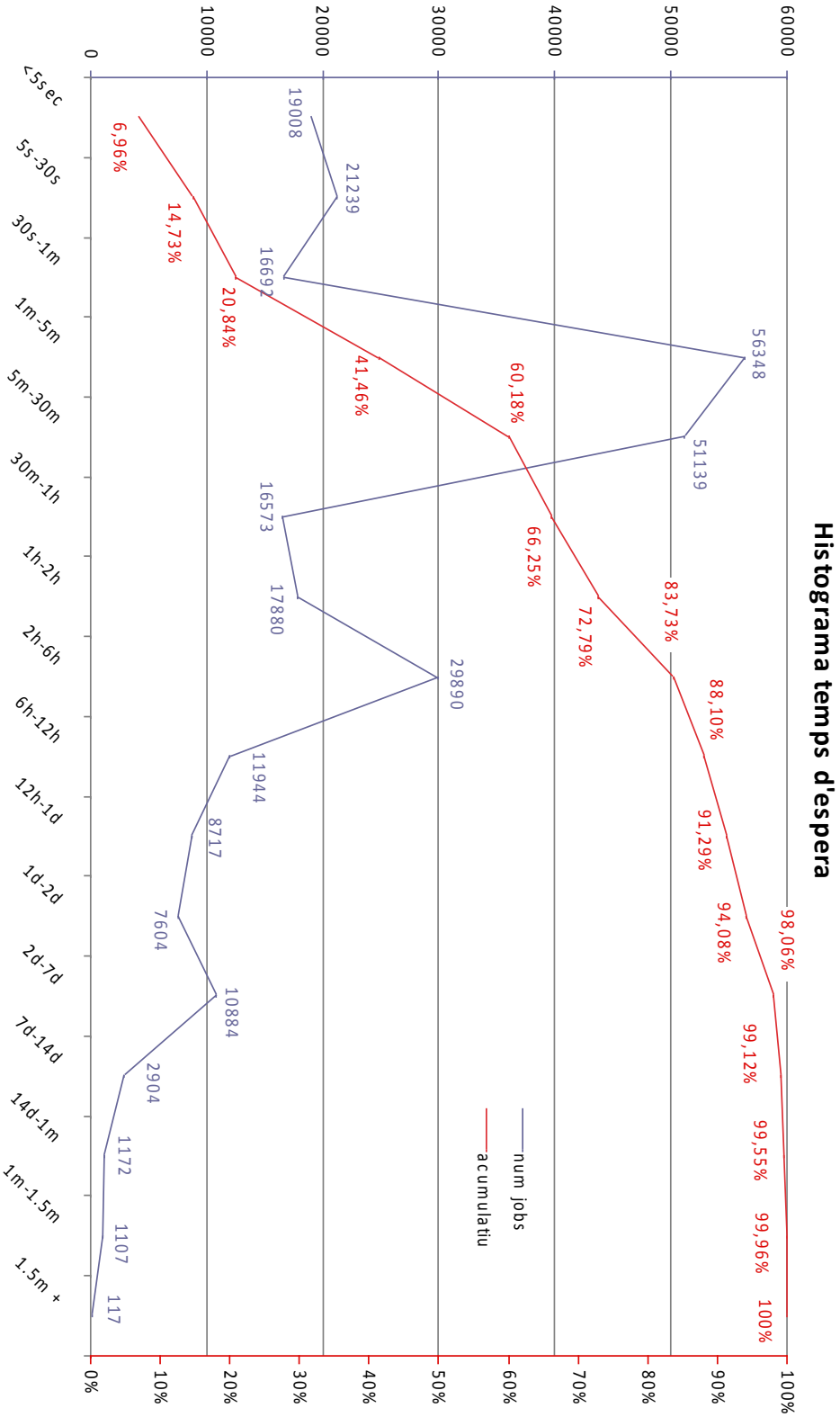

Figura 6.15 – Histograma del temps d'espera, en nombre de jobs i % acumulat.

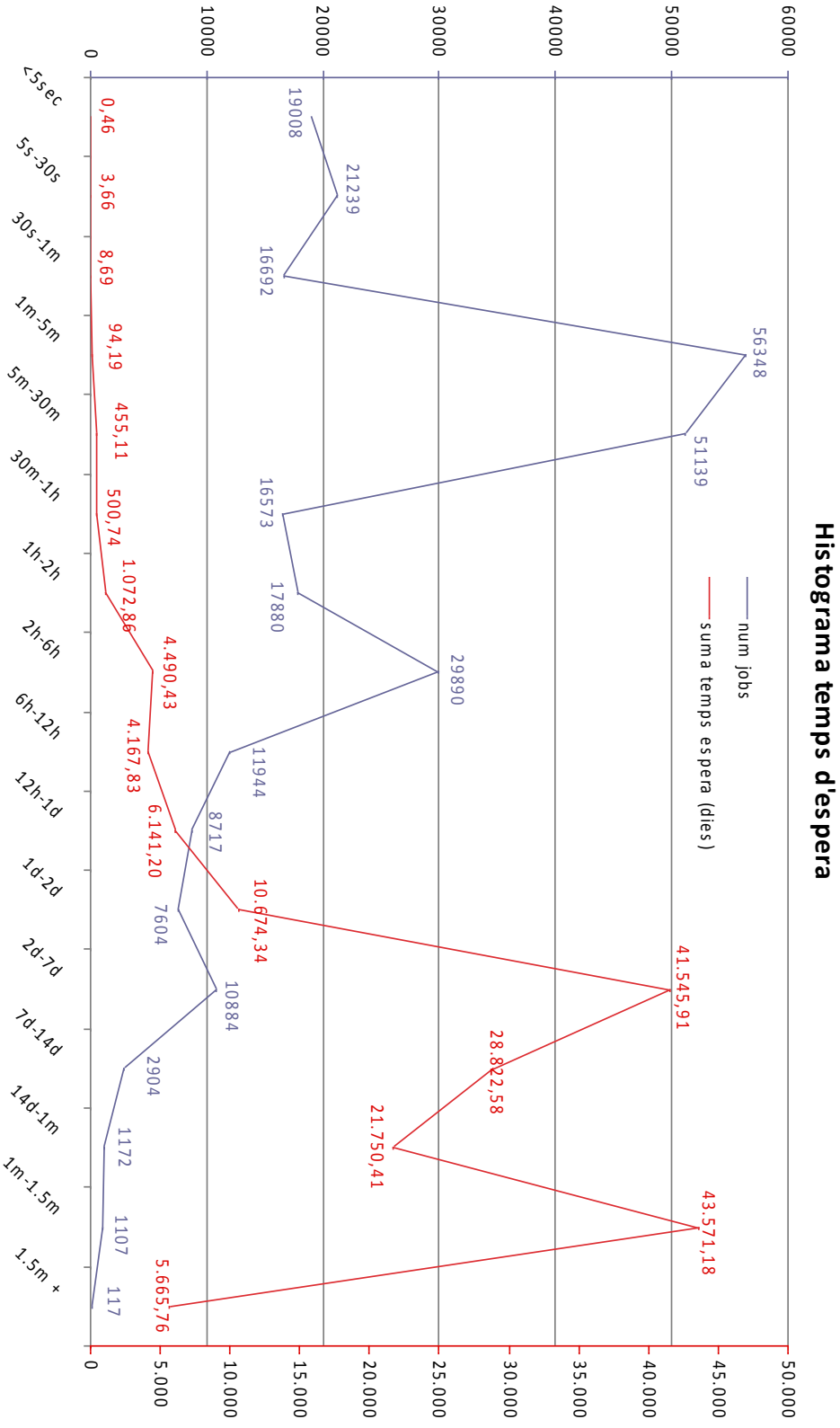

Figura 6.16 – Histograma del temps d'espera, en nombre de jobs i suma de dies en espera.

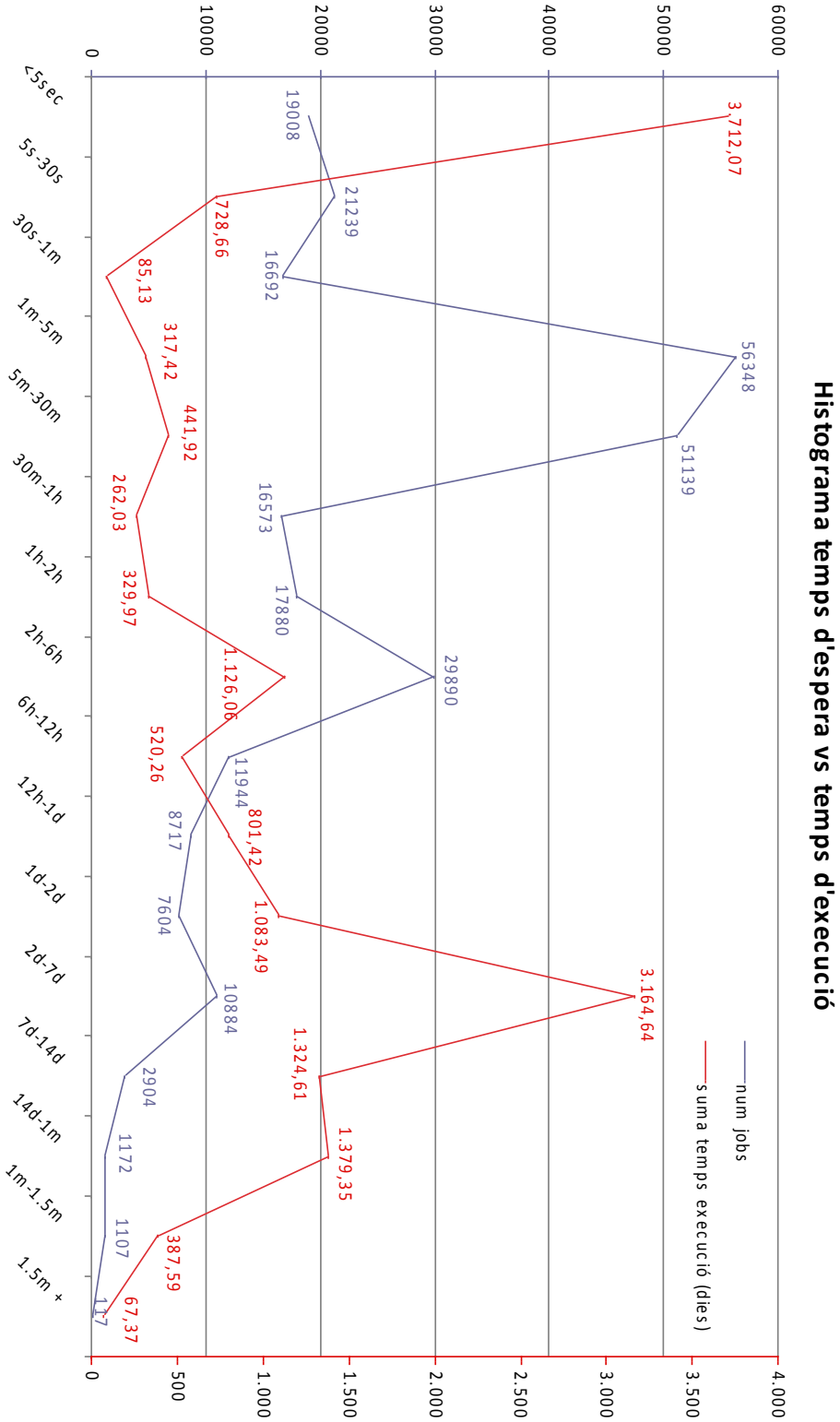

Figura 6.17 – Histograma del temps d'espera, en nombre de jobs i suma de dies.

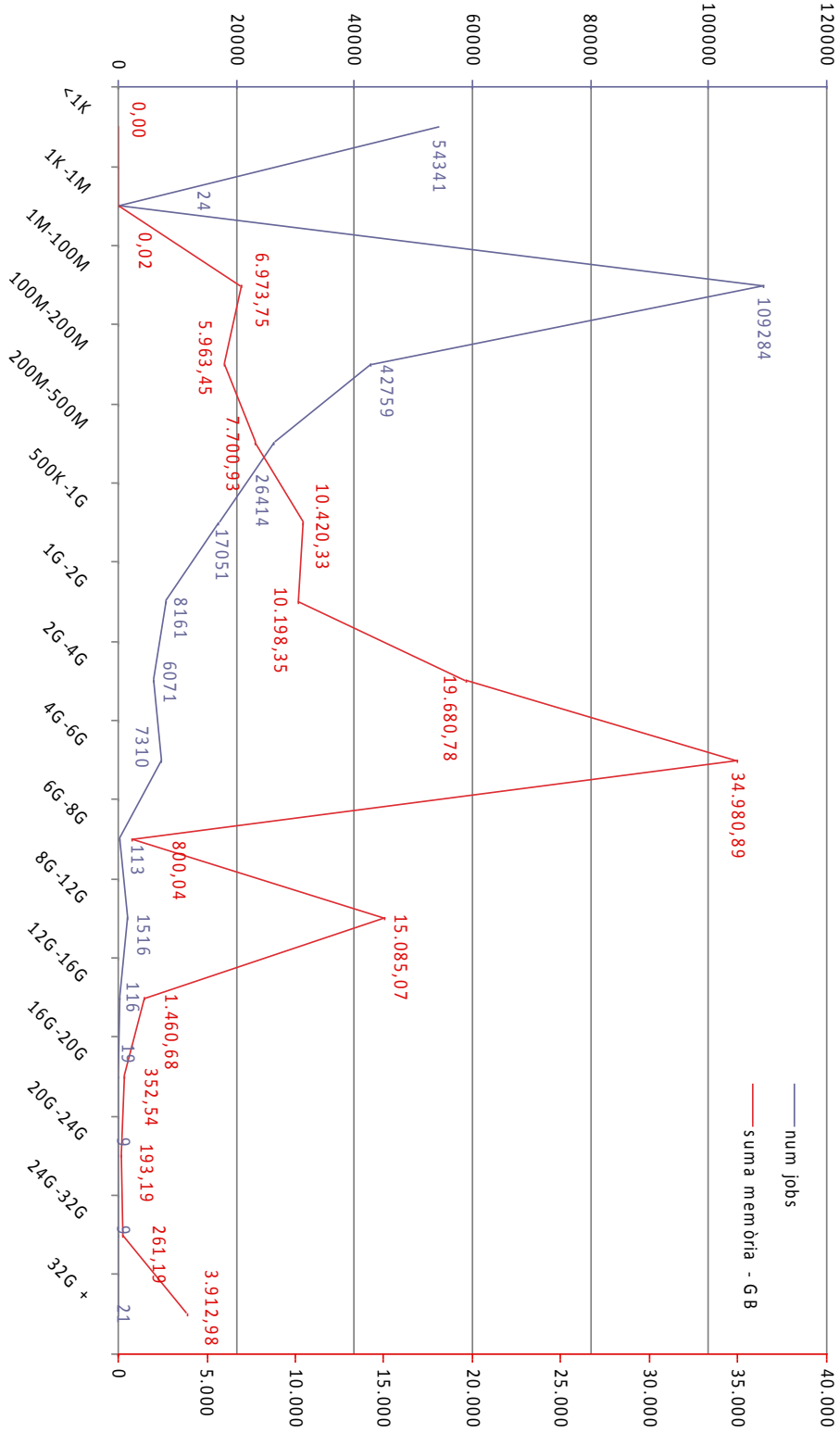

Figura 6.18 – Histograma del màxim de memòria usada, en nombre de jobs i suma de memòria.

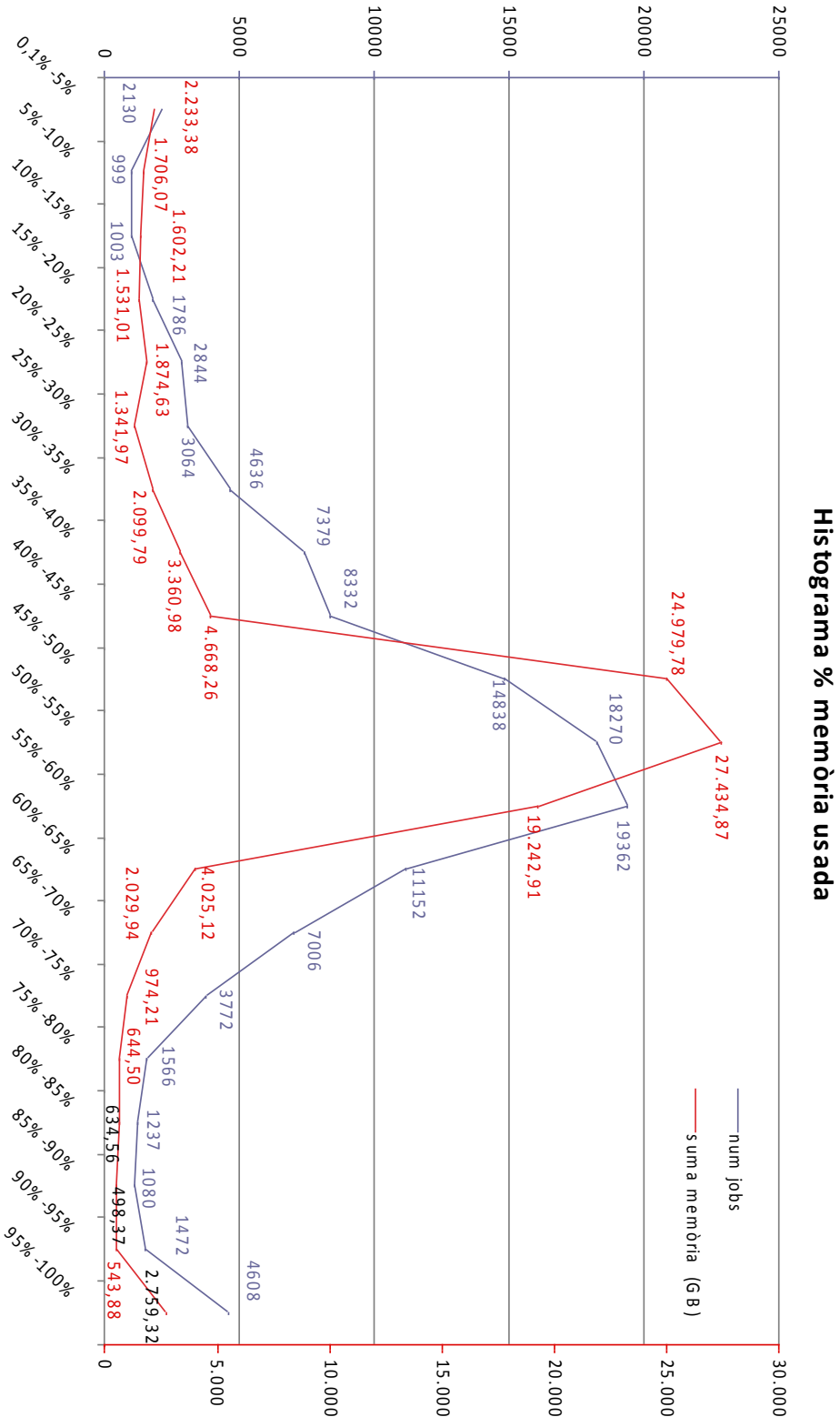

Figura 6.19 – Histograma del % de memòria usada respecte a l'àrea, en nombre de jobs i suma de memòria.

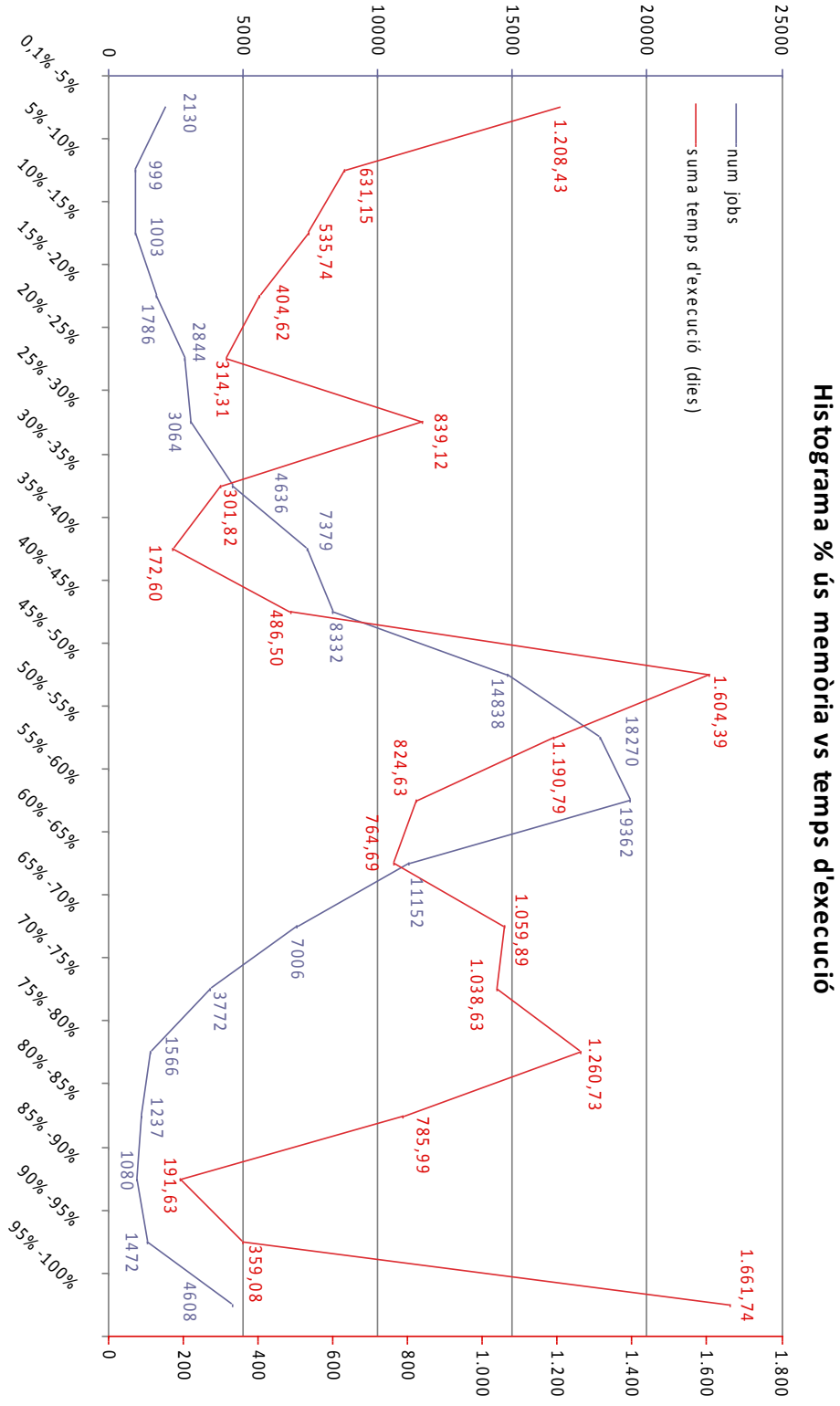

Figura 6.20 – Histograma del % de memòria usada respecte a l'àrea, en nombre de jobs i suma de temps d'execució.

Ús usuaris

Com podem observar (*figures 6.21 i 6.22*), el nombre d'usuaris del sistema ha anat creixent significativament al llarg del temps. Tot i això, veiem que hi ha una clara presència d'usuaris "avançats", els quals en fan ús al llarg del temps, i usuaris "puntualment intensius", els quals apareixen puntualment fent un ús important dels recursos durant un breu període.

També podem veure (*figures 6.23 i 6.24*) el comportament dels usuaris des de la posada en marxa de la política de penalització per mal ús de la reserva de memòria. Totes dues gràfiques ens presenten als usuaris ordenats decreixentment segons el nombre de jobs executats. Gràcies a això, podem apreciar l'efectivitat de la política, ja que veiem com els usuaris aprenen a fer un millor ús del sistema a mesura que l'utilitzen, reduint el nombre de jobs que reserven molta més (un 50% més) memòria de la que realment necessiten i els jobs "uninformed" (processos executats sense preocupar-se de demanar la memòria).

Finalment, veiem (*figura 6.25*) el "guany" de temps que ha suposat per als usuaris l'ús del clúster, degut principalment a la capacitat d'executar múltiples processos simultàniament. Els resultats d'aquesta gràfica ens donen una idea de la importància que està prenent el sistema dins de l'institut. En alguns casos els usuaris han arribat a executar l'equivalent a 11 anys de càlcul, en poc més de 6 mesos!

L'efecte real d'aquest ús el podem comprovar a l'*Annex VI, llistat de publicacions que han fet ús del sistema*.

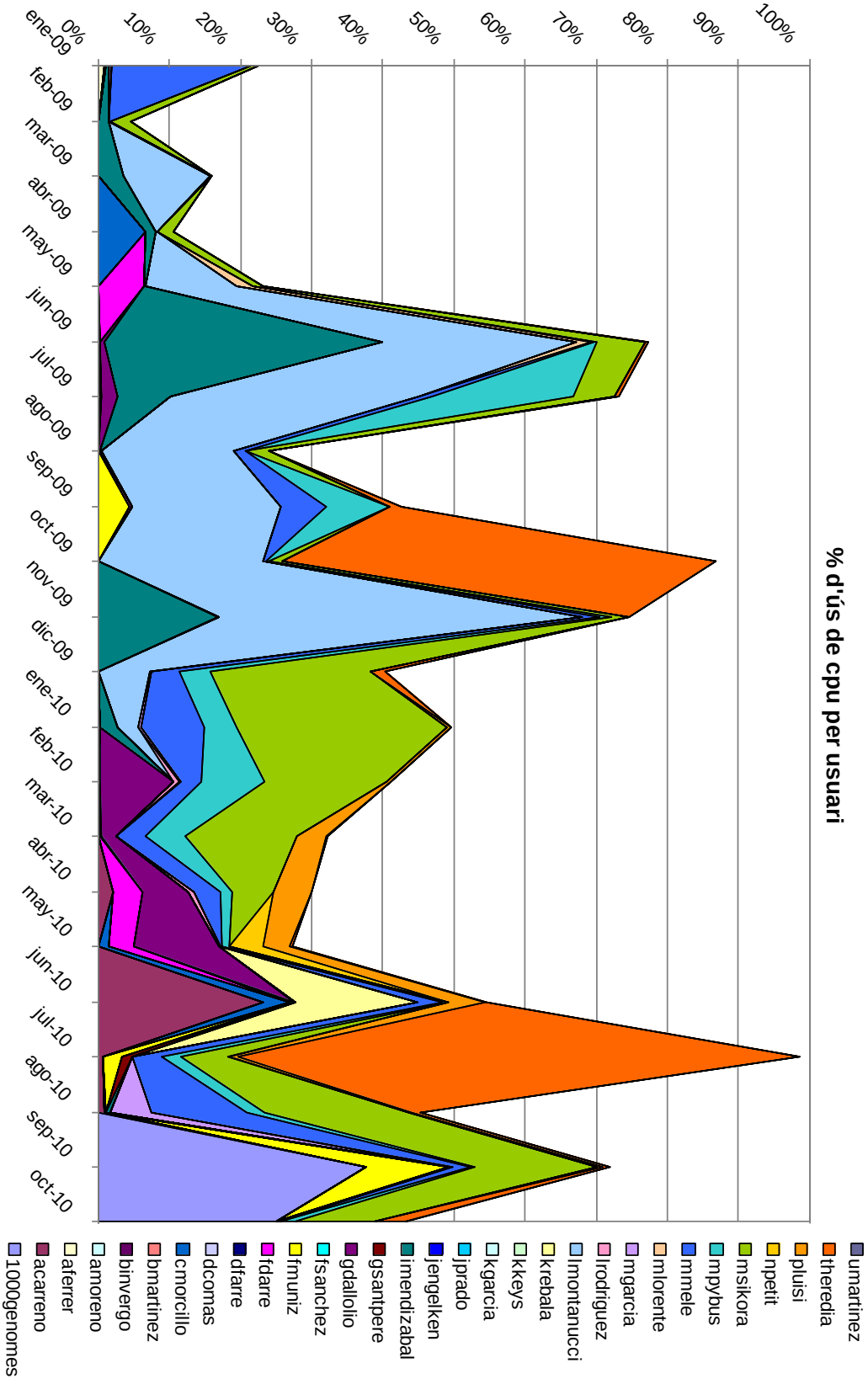

Figura 6.21 - Ús de cpu mensual per usuari, en %.

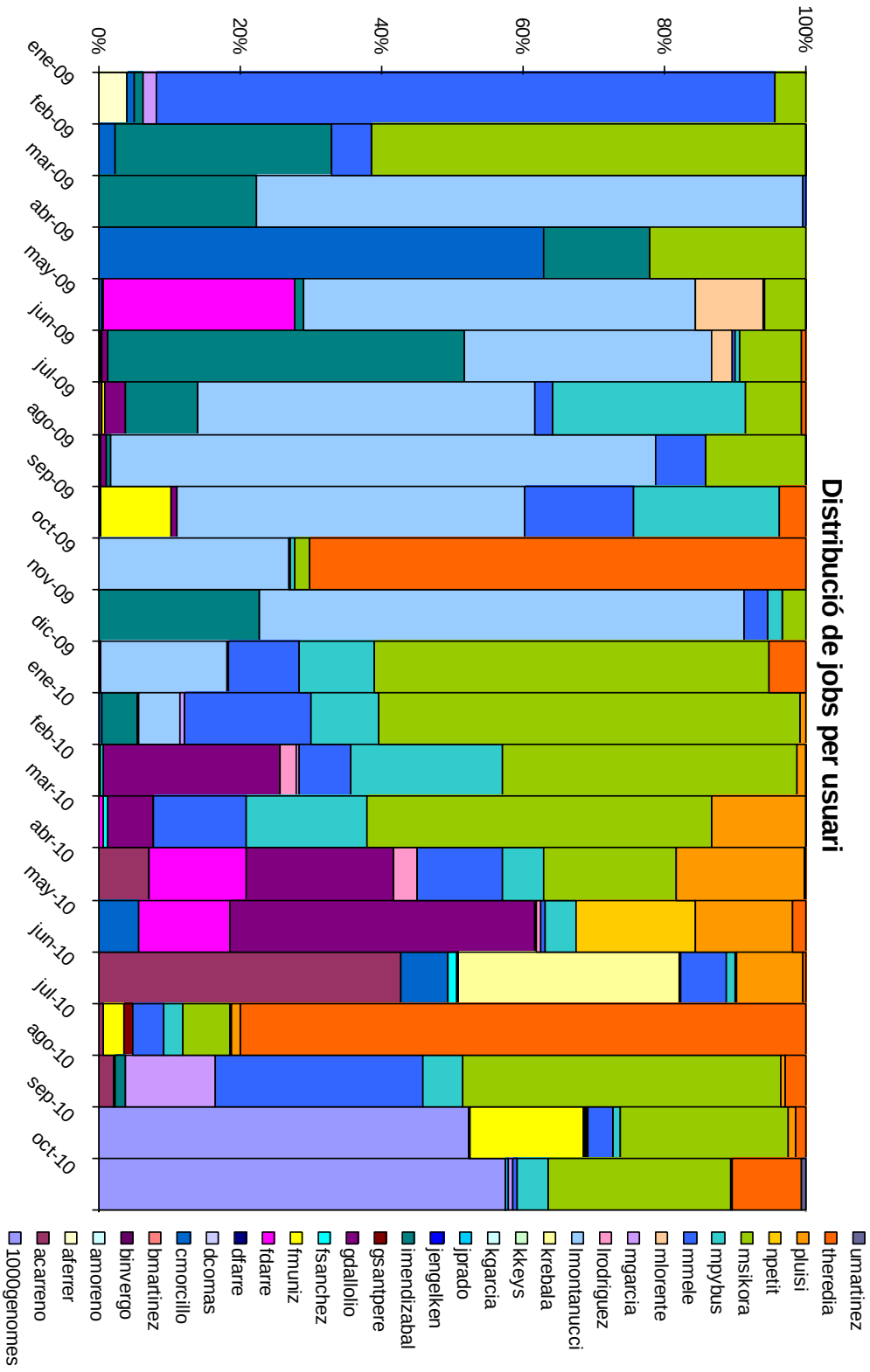

Figura 6.22 – Distribució dels jobs per usuari, en % respecte el total de jobs del mes.

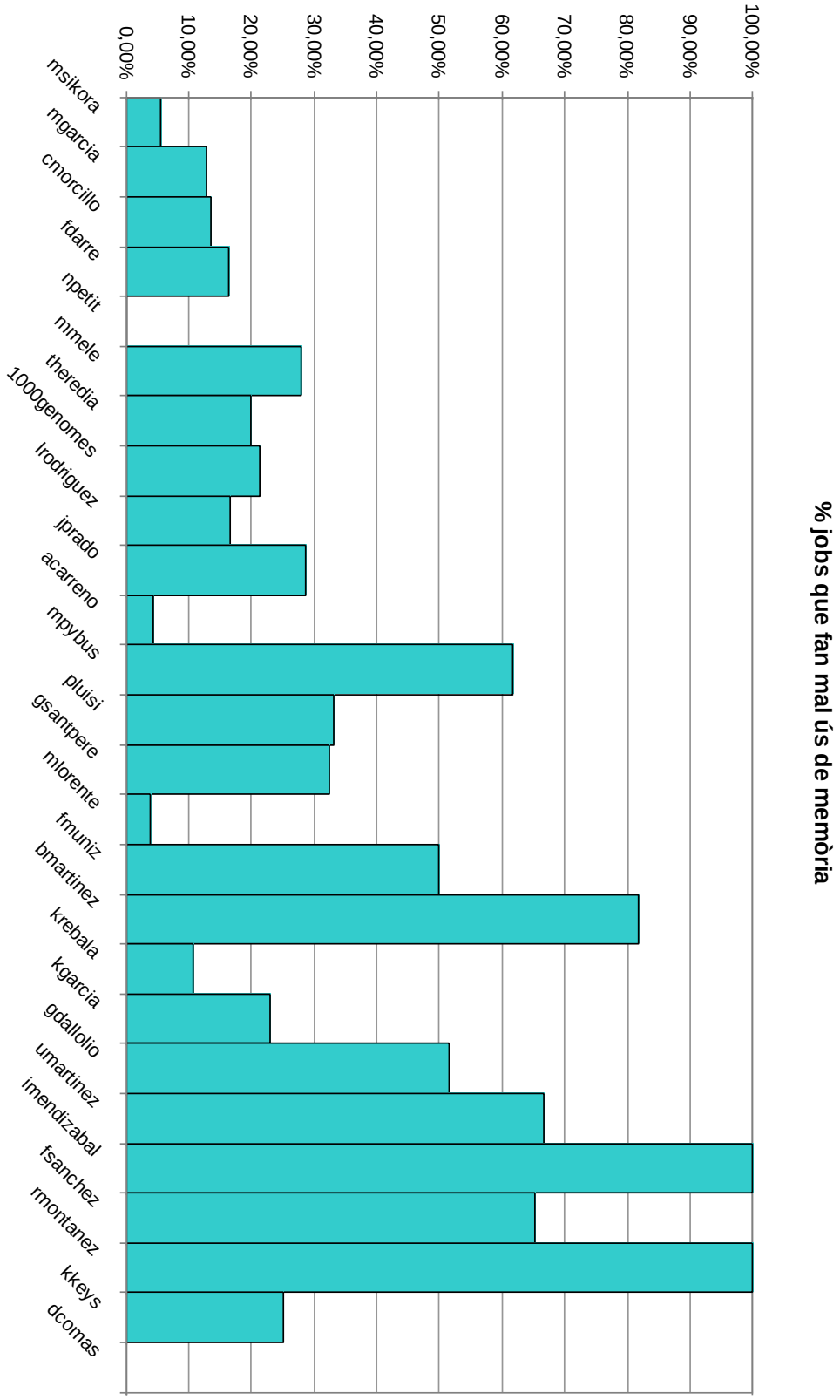

Figura 6.23 – Percentatge de jobs que han fet un mal ús de memòria per usuari.

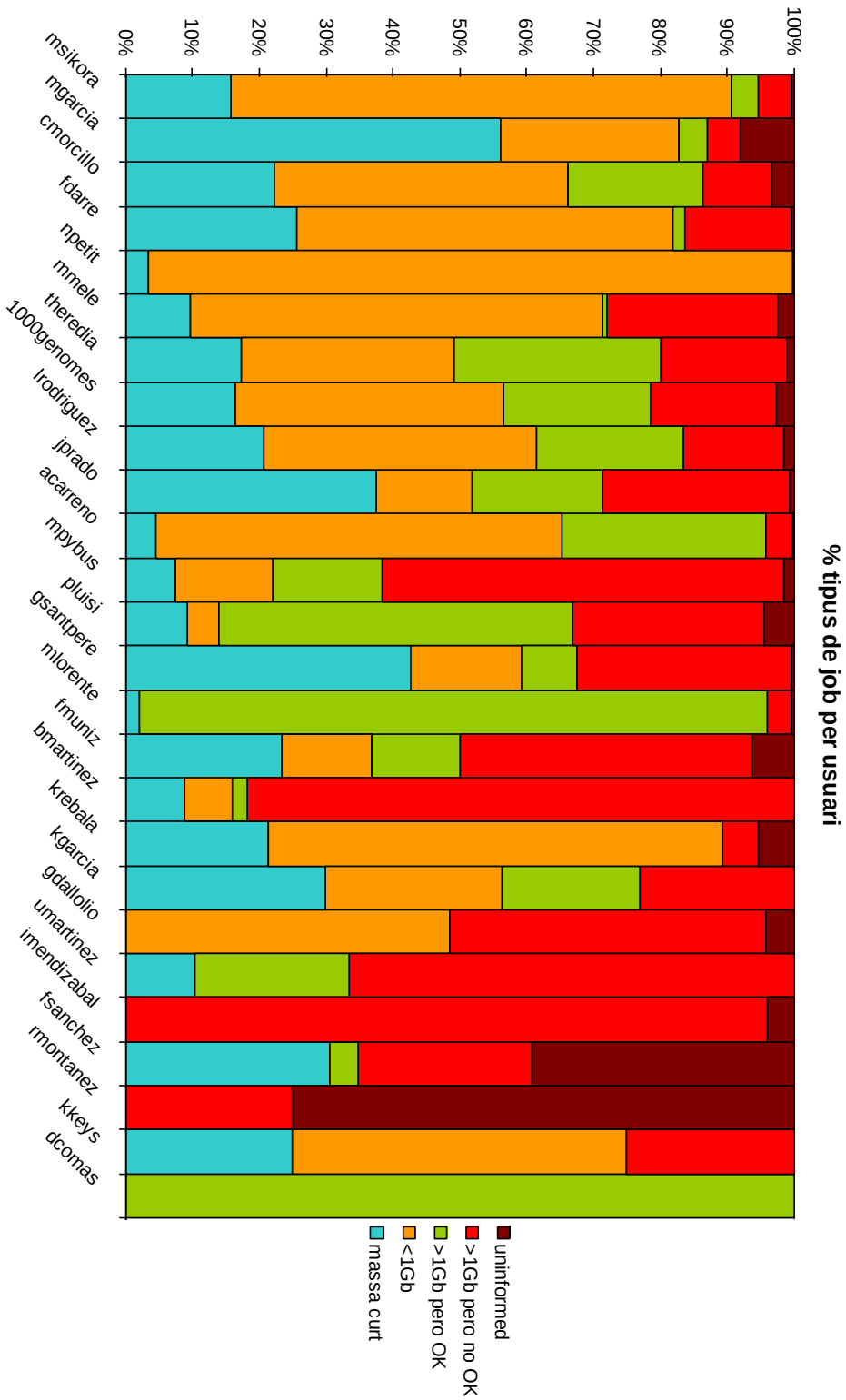

Figura 6.24 – Percentatge de jobs per usuari, classificats segons l'ús de memòria.

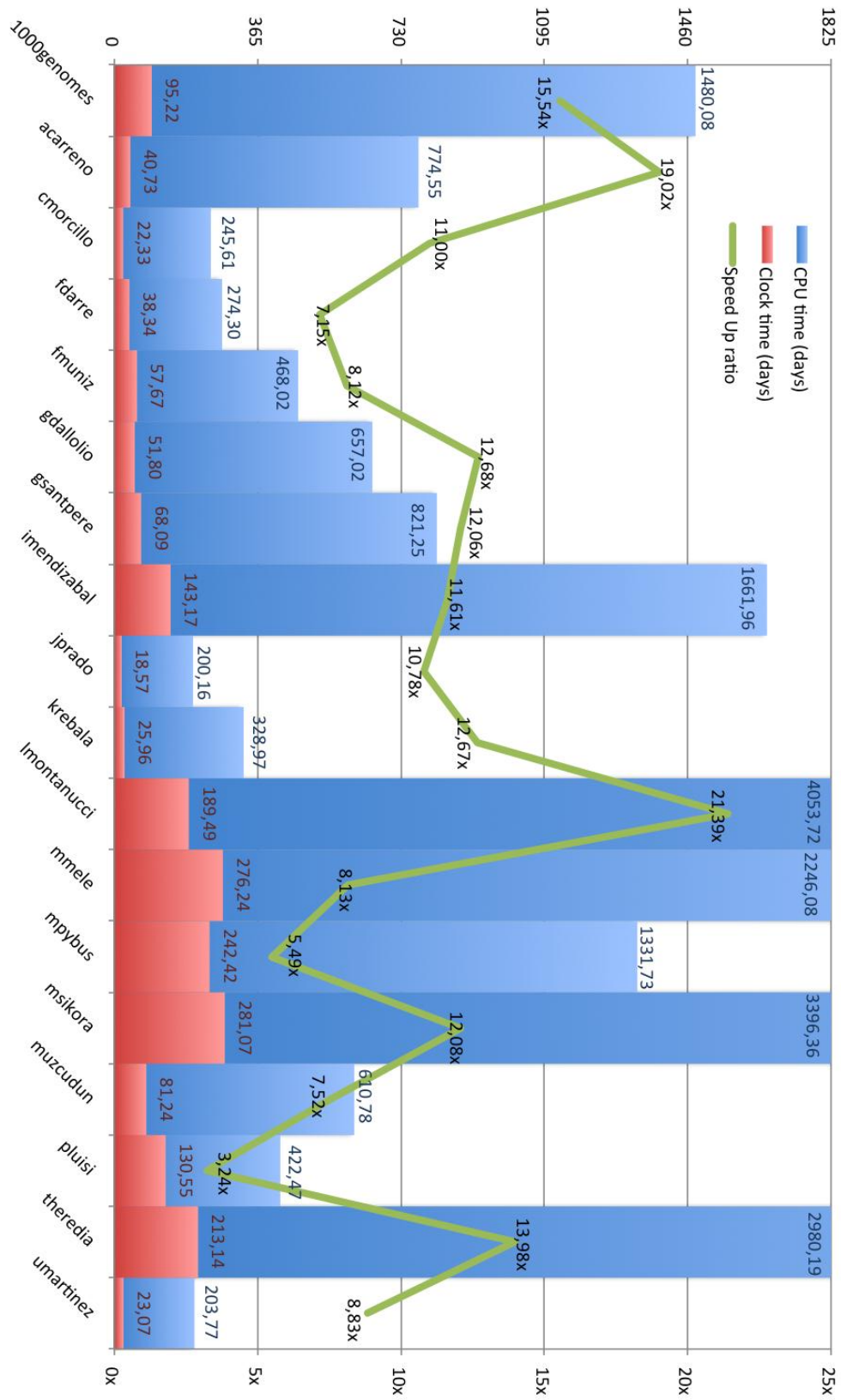

Figura 6.25 – Comparativa per usuari del temps de càlcul utilitzat respecte al temps real.

Actuacions

Des de la posada en marxa del sistema ha calgut realitzar diverses actuacions per tal de modificar i millorar el seu funcionament, així com adaptar-nos als canvis i nous requisits que han anat sorgint amb el temps.

Usuaris:
Inicialment, quan vam començar la fase de proves, el sistema comptava amb 7 usuaris. Posteriorment, quan vam implementar el sistema de backups remots, vam decidir crear un usuari per a cada membre del departament per a que podessin fer-lo servir. Actualment tenim 80 usuaris classificats de la següent manera:

- 59 usuaris actuals:
    - 40 usuaris del clúster de càlcul.
        - 24 dels quals són usuaris habituals.
    - 12 usuaris utilitzen el sistema de backup (la majoria treballen amb el mateix investigador principal).
    - 16 usuaris que no donen cap ús al sistema (la majoria personal de laboratori o IP's).
- 21 usuaris antics:
    - 13 dels quals van ser usuaris actius del clúster.
    - En conservem els backups de les dades científiques de 6 d'ells.

Instal·lacions:
Actualment tenim instal·lats 75 programes científics per a ús de tots els usuaris del sistema.

Incidències de hardware:
Hem tingut diverses incidències de sistema. Quan un element de hardware falla i el sistema gestor del xassís detecta un error que no especifica exactament el seu origen, el protocol estàndard d'IBM consisteix en canviar la placa base. Degut a això hem tingut 7 actuacions de canvi de placa base. També hem tingut una memòria RAM defectuosa i una incidència en una CPU. Addicionalment hem tingut 3 incidències de disc dur que van ser convenientment substituïts.

Formació i Cursos

Per tal de facilitar la introducció i ús del sistema als usuaris, hem creat un breu tutorial d'ús del sistema per a nous usuaris. També hem creat diversos manuals explicant com utilitzar les funcionalitats addicionals del clúster, com el sistema de backups remots o un tutorial de creació de repositoris web per a publicar dades mitjançant apache. Podreu trobar més informació sobre aquests tutorials a l'*Annex V, tutorials i documents de suport a l'usuari*.

També hem fet diversos cursos i presentacions del sistema:

- Seminari sobre bases de dades: avantatges, ús de bases de dades públiques i introducció al SQL.
- Dos pòsters de presentació del clúster als nous membres, durant els dos retreats conjunts realitzats des de la creació de l'Institut de Biologia Evolutiva.
- Reunions mensuals amb els usuaris per a explicar-los novetats i resoldre dubtes.
- Seminari introductori del sistema per a tots els usuaris.

Finalment, per tal que jo comprengués el rerefons biològic de tot el sistema i així poder comprendre millor les necessitats dels usuaris i el sistema, he cursat diverses assignatures de la carrera de biologia a la UPF, entre elles comprensió biològica dels humans, ecologia, evolució i genètica, així com l'assistència a multitud de seminaris científics.

Perspectives de futur

Durant el temps transcorregut des de la planificació inicial del sistema i la seva posada en marxa, fins a avui dia, hem anat aprenent i coneixent el funcionament i necessitats

reals del sistema i dels seus usuaris. Això, conjuntament amb l'evolució de les necessitats del departament i dels seus projectes ens ha portat a una evolució contínua del sistema.

Ampliacions ja realitzades

Degut a les necessitats dels usuaris, vam veure necessària una ampliació de memòria a tots els nodes, ja que, inicialment, disposàvem només de 1,125GB per procés (9GB-node / 8 cores-node), i va arribar un moment en que això va resultar insuficient. Per aquest motiu vam incrementar la memòria dels nodes fins a una mitjana de 23GB.

Pel que fa al disc, les necessitats dels usuaris i projectes han anat creixent contínuament. Això ha fet necessàries (moltes vegades amb urgència) diverses ampliacions de disc. Així, la cabina de disc inicial (IBM DS4200 amb 16 slots de disc) va passar de tenir inicialment 11 discs de 500GB, a omplir tots els slots de disc. Més endavant això va tornar a quedar-se curt i ens vam veure obligats a comprar una safata d'ampliació de la cabina (de 16 slots també), amb 3 discs d' 1TB cadascun. Posteriorment vam ampliar a 7 discs i finalment vam acabar omplint-la amb 16. Això ens dóna actualment un total de 16 discs de 500GB i 16 de 1TB, dels quals el clúster de càlcul ara mateix en serveix 7,5TB (5 d'ells com a *scratch*), i la resta estan distribuïts entre redundància i particions especials per a emmagatzemar dades de diversos projectes.

Ampliacions futures

Inicialment el sistema es va dimensionar per a ser usat per la Unitat de Biologia Evolutiva de la Universitat Pompeu Fabra, però, posteriorment es va decidir fusionar aquest amb el Departament de Fisiologia i Biologia Molecular del CSIC, creant l'Institut de Biologia Evolutiva (UPF-CSIC). Aquesta fusió duplicava efectivament el nombre d'usuaris potencials del sistema. La integració s'està realitzant progressivament, i ara mateix 8 dels usuaris del sistema provenen d'aquesta unió. Per aquest motiu, es va decidir planejar una ampliació del clúster, en disc i en nombre de nodes. L'ampliació de disc s'ha fet efectiva ja amb la compra d'una nova cabina de disc, la qual, a part d'oferir-nos més espai, té funcionalitats de servidor NAS i CIFS. Això ens permetrà derivar en ella les tasques de servei de disc per xarxa i alliberar així part dels recursos del clúster d'alta disponibilitat, els quals podrem utilitzar per altres tasques. Pel que respecta a

l'ampliació de nodes, degut a la crisi econòmica encara no s'ha materialitzat, tot i que segueix prevista però sense una data clara.

Per altra banda, aquest darrer curs s'ha incorporat a l'institut un nou investigador principal que, un cop hagi acabat de formar el seu grup, tindrà unes grans necessitats computacionals. Aquestes necessitats incrementades suposaran una injecció de pressupost pel sistema, gracies a la qual preveiem ampliar el nombre de nodes de càlcul fins a, com a mínim, doblar-lo. Això, però, comportarà la necessitat d'establir polítiques de priorització de treballs al clúster, depenent del projecte al que pertanyin.

Finalment, tenim previstes diverses actuacions d'actualització i millora del sistema. Aquestes es realitzaran quan es faci efectiva l'actualització dels nous nodes, ja que fins ara no les hem pogut dur a terme degut al constant ús del sistema en producció i a la manca d'un ambient de test amb les mateixes condicions que en producció. Entre aquestes actuacions hi destaquen:

- Actualització dels sistema operatiu tant del clúster d'alta disponibilitat com del de càlcul.
- Incorporació d'un tercer node al clúster d'alta disponibilitat per a trencar la falta de quòrum en cas de caiguda de la xarxa.
- Creació d'un segon node de càlcul en alta disponibilitat preparat per a actuar com a front-end en cas de caiguda del principal.

Un aspecte important d'aquestes futures actualitzacions és el futur del sistema gestor de cues, ja que la compra de SUN per part d'Oracle ha compromès el futur de Sun Grid Engine, i sembla que la comunitat de codi lliure està optant per migrar cap a altres sistemes gestors de cues.

Valoració Econòmica

De cara a realitzar la valoració econòmica del projecte, caldrà tenir en compte el cost tant de maquinari i programari, com de les hores de treball, així com les diferents etapes que ha tingut el projecte.

Configuració inicial:

Dividirem el cost el sistema entre el maquinari i llicències de software, i el cost de personal de manteniment del sistema. El cost del muntatge del maquinari ve inclòs dins dels pressupostos del sistema, i les hores de tècnic de comunicacions inclouen totes les tasques de connexió i adaptació a la xarxa de la universitat realitzades pels departaments de comunicacions i suport de projectes de recerca.

Maquinari i programari	
Concepte	Subtotal
Cabina de disc IBM DS4200	19.906,97 €
Xassís de blades	68.318,49 €
Llicències de software	180,00 €
Total	**88.405,46 €**

Personal			
Concepte	Hores	Cost/hora	Subtotal
Muntatge del hardware			2.146 €
Configuració inicial del clúster d'alta disponibilitat	500	8	4.000 €
Configuració inicial del clúster de càlcul	200	8	1.600 €
Tècnic de comunicacions	30	40	1.200 €
Total			**8.946 €**

Ampliacions del sistema:

Degut a la natura tant del sistema com de les nostres necessitats i les nostres fonts de finançament, avaluarem les ampliacions del sistema de manera conjunta. Pel que fa a les hores de feina, les unificarem també, però tenint en compte que la majoria del temps de muntatge i instal·lació està dedicat a la segona ampliació de disc, ja que les altres ampliacions resulten força senzilles al tractar-se de dispositius de connexió en calent o força fàcils de connectar.

Maquinari i programari	
Concepte	Subtotal
1ª ampliació de disc (5 discs de 500GB)	2.175,00 €
2ª ampliació de disc (safata d'expansió + 3 discs d'1TB)	10.898,20 €
3ª ampliació de disc (4 discs d'1TB)	2.732,96 €
4ª ampliació de disc (9 discs d'1TB)	6.149,16 €
Ampliació de RAM	6.672,32 €
Total	**28.627,64 €**

Personal			
Concepte	Hores	Cost/hora	Subtotal
Muntatge del hardware	20	40	800 €
Configuració del sistema per a adaptar-lo a les noves configuracions	230	8	1.840 €
Tècnic de suport	30	40	1.200 €
Total			**3.840 €**

Manteniment del sistema:

Dins del manteniment del sistema hi inclourem diversos elements, com són el cost del contracte de manteniment i substitució de peces del fabricant, les hores dedicades al manteniment tècnic del sistema, així com les dedicades a la instal·lació de programes, llibreries o gestió d'usuaris. Tots aquests costos seran considerats anualment.

Maquinari i programari	
Concepte	Subtotal
Contracte de manteniment	1.679,10 €/3 anys
Lloguer de l'espai al CPD	0 €/any
Consum elèctric	4,8 kW
Total	**559,70 €/any**

Personal			
Concepte	Hores anuals	Cost/hora	Subtotal
Incidències	85	8	680 €
Tècnic de suport	25	40	1.000 €
Manteniment tècnic	60	8	480 €
Administració del sistema	200	8	1.600 €
Total			**3.840 €/any**

Cost total:

Així doncs, podem veure que el cost total del sistema ha estat de:

Cost total	
Concepte	Subtotal
Despeses materials	117.033,10 €
Despeses de personal	12.786,00 €
Total	**129.819,10 €**
Cost de manteniment anual	+ 4399,70 €/any

Glossari

Array (de discs):
Conjunt de discs físics que formaran un volum en RAID.

Batch:
Conjunt de jobs similars enviats al sistema simultàniament. Sol tractar-se de diverses execucions d'un mateix procés, però utilitzant diversos conjunts de dades d'entrada, paràmetres d'execució diferents, o execucions idèntiques de processos en que hi intervé un component d'atzar.

Cabina de disc:
Nom comú que reben els sistemes de hardware d'emmagatzemament de disc dedicat. Consisteixen bàsicament en un maquinari amb capacitat per a muntar diversos discs i servir-los utilitzant diferents mecanismes. El seu principal avantatge és la seguretat que ofereixen al disposar de sistemes de creació de RAIDs, discs de *hot-spare*, recuperació automàtica de dades, alertes d'incidències de disc, ...

Clúster / *Grid*:
Tipus de sistema de computació basat en l'ús d'un conjunt d'ordinadors individuals i independents, però interconnectats d'alguna manera entre ells. Aquest sistemes utilitzen diversos tipus de software (siguin sistemes operatius, *middlewares*, virtualitzadors o sistemes de gestió de recursos) amb l'objectiu específic de poder tractar totes aquestes màquines independents com si d'un únic sistema es tractés per tal de poder-hi executar càlculs complexos que no podrien satisfer cap dels sistemes individualment.

Cua:
Abstracció que utilitzen els sistemes de gestió de recursos per a controlar quins processos estan en execució i quins en espera. No cal que sigui estrictament una cua en el sentit "*First In, First Out*".

Fibre Channel:
Tecnologia i protocol de comunicació utilitzat principalment en la comunicació amb sistemes de disc. Tot i el seu nom, pot funcionar tant sobre cables de fibra òptica com sobre cables de coure trenat.

Front-end:
Node d'un clúster que actuarà com a punt d'entrada i interactuació dels usuaris amb el sistema. Aquest node pot o no estar destinat també a l'execució de càlculs.

Job:
Procés o treball que l'usuari executarà al sistema. Pot tractar-se de qualsevol tipus de procés, sigui un script realitzat pels usuaris, un binari compilat, o una comanda del sistema excessivament llarga i costosa. L'únic requisit principal és que sigui executable en *background* i no requereixi d'intervenció humana per a completar-se.

LVM:
Inicials de *Logical Volume Manager*. Es tracta d'una capa de software que genera una abstracció entre els volums físics i lògics del sistema, permetent una flexibilitat molt més gran que els sistemes de particionat tradicionals.

NFS:
Inicials de *Network File System*. Es tracta d'un sistema de fitxers en xarxa que permet que diversos clients puguin muntar un volum remot per a accedir a les dades contingudes al node servidor.

Node:
Ordinador membre d'un clúster.

Pipeline:
Conjunt de processos que tenen una forta dependència de dades entre ells i que s'han d'executar de manera seqüencial entre ells. Poden tractar-se com un únic job (l'usuari crea tota la seqüència i gestió de la pipeline dins d'un script), o com a diversos jobs, delegant en el sistema gestor de recursos tot el control d'esperes, dependències i semàfors.

SGE:

Inicials de *Sun Grid Engine*, el sistema gestor de recursos i cues que utilitzarem.

Sistema Gestor de Recursos / *Distributed Resource Manager*:

Software especialitzat en la distribució de l'ús de recursos d'un sistema (memòria, temps de procés, disc, ...), entre els diversos processos a executar-se en el sistema. Solen ser l'element principal de control i distribució en sistemes clúster.

Agraïments

La realització de tot aquest projecte no hagués estat possible sense l'inestimable suport dels meus companys Angel Carreño Torres, de l'Institut de Biologia Evolutiva, Carles Perarnau i Sabés i Marc Esteve i Crespo, del departament de Suport a la Recerca de la Universitat Pompeu Fabra, Núria Reixach i Pastoret, de la Unitat d'Informàtica de l'Àrea Mar de la Universitat Pompeu Fabra, així com també l'ajut rebut per l'Alfons González i Pauner, de l'Institut Municipal d'Investigació Mèdica, l'Ismael Pérez Laguna, del servei de Comunicacions i l'Iván Jiménez Roda del servei d'informàtica del Campus Poblenou de la Universitat Pompeu Fabra.

Bibliografia

Per tal de realitzar aquest projecte ha estat necessària la consulta de diverses fonts bibliogràfiques:

Manuals d'instal·lació:

- Blade Network, Inc.: *BladeOS Application Guide for BNT Layer 2/3 GbE Switch Module for IBM BladeCenter*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/bmd00174.pdf

- IBM, Corp.: *BladeCenter H Type 8852 – Installation and User's Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/81y1106.pdf

- IBM, Corp.: *BladeCenter Advanced Management Module – User's Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/bladecenter/16y6036.pdf

- IBM, Corp.: *QLogic 8Gb Intelligent Pass-thru Module for IBM BladeCenter and QLogic 20-Port 8Gb SAN Switch Module for IBM BladeCenter – Installation and User's Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/44r5237.pdf

- IBM, Corp.: *BladeCenter HS21 Type 7995 – Installation and User's Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/44w1496.pdf

- IBM, Corp.: *QLogic 4Gb Fibre Channel Expansion Card (CFFv) for IBM BladeCenter – Installation and User's Guide*.

http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/42c4875.pdf

- IBM, Corp.: *IBM System Storage DS4200 Express Storage Subsystem Fibre Channel Cabling Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/gc27204900.pdf

- IBM, Corp.: *IBM System Storage DS Storage Manager Version 10 – Installation and Host Support Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/60y1739.pdf

- IBM, Corp.: *IBM System Storage DS4200 Express Storage Subsystem – Installation, User's and Maintenance Guide*. http://download.boulder.ibm.com/ibmdl/pub/systems/support/system_x_pdf/gc27204802.pdf

Documentació de software:

- Novell, Inc.: *Documentació d'instal·lació, configuració i suport de SUSE Linux Enterprise Server*. http://www.novell.com/documentation/sles10/

- Rocks-cluster: *Rocks-cluster 5.0 – Base Roll Documentation Guide*. http://www.rocksclusters.org/roll-documentation/base/5.0/roll-base-usersguide.pdf

- Novell, Inc.: *Linux High Availability Advanced Technical Training – Lecture Manual*.

- The Apache Software Foundation: *Apache HTTP Server Version 2.2 Documentation*.
  http://httpd.apache.org/docs/2.2/

- Oracle, Corp.: *MySQL 5.1 Reference Manual*.
  http://downloads.mysql.com/docs/refman-5.1-en.a4.pdf

- A. J. Lewis: *Logical Volume Manager How to*.
  http://tldp.org/HOWTO/LVM-HOWTO/

- The Samba Team: *The Official Samba 3.5.x HOWTO and Reference Guide*.
  http://www.samba.org/samba/docs/man/Samba-HOWTO-Collection/
  http://www.samba.org/samba/docs/

- Christopher Smith: *Linux NFS-HOWTO*.
  http://nfs.sourceforge.net/nfs-howto/

- *Network Time Protocol (NTP) Documentation*.
  http://www.eecis.udel.edu/~mills/ntp/html/index.html

- Nagios Enterprises: *Nagios Core 3.x Documentation*.
  http://nagios.sourceforge.net/docs/nagioscore-3-en.pdf

- *Ganglia documentation wiki*.
  http://sourceforge.net/apps/trac/ganglia

- The OpenLDAP Foundation: *OpenLDAP 2.3 Administration Guide*.
  http://www.openldap.org/doc/admin23/OpenLDAP-Admin-Guide.pdf

- Oracle, Corp.: *Sun Grid Engine Documentation Home*.
  http://wikis.sun.com/display/GridEngine/Home

Altres fonts d'informació:

- Wikipedia: http://www.wikipedia.org/
- Llista de distribució de Sun Grid Engine: users@gridengine.sunsource.net (desapareguda per la compra de SUN per part d'Oracle).
- Llista de distribució de Rocks-cluster: npaci-rocks-discussion@sdsc.edu
- Llista de distribució de NFS: linux-nfs@vger.kernel.org
- Fòrums de Linux-Questions: http://www.linuxquestions.org/
- ... i desenes de webs i fòrums cercant informació i documentació durant incidències o instal·lacions de nou software.

Annexos

Índex

Annex I, pressupostos

A continuació mostrem un detall dels pressupostos amb els que hem construït el sistema:

Pressupost Inicial

BladeCenter			
Concepte	Qtt	Preu unitari	Subtotal
IBM eServer BladeCenter(tm) H Chassis with 2x2900W PSU	1	3.769,74 €	3.769,74 €
IBM BladeCenter(tm) H 2900W AC Power Supply Modules	1	699,22 €	699,22 €
Nortel Networks Layer 2/3 Copper GbE Switch Module for BladeCenter	2	1.269,85 €	2.539,70 €
QLogic(R) 10-port 4 Gb SAN Switch Module BladeCenter	2	3.105,89 €	6.211,78 €
4 Gbps SW SFP Transceiver 4 Pack	2	316,67 €	633,34 €
2.8m, 200-240V, Triple 16A IEC 320-C20	2	65,08 €	130,16 €
DPI 63amp/250V Front-end PDU with IEC 309 2P+Gnd	2	390,48 €	780,96 €
Blades HS21XM			
HS21 XM, Xeon Quad Core E5345 2.33GHz/1333MHz/8MB L2, 2x512MB, O/Bay SAS	11	1.719,66 €	18.916,26 €
Intel Xeon Quad Core Processor Model E5345 80w 2.33GHz/1333MHz/8MB L2	11	619,85 €	6.818,35 €
4GB (2x2GB) PC2-5300 CL5 ECC DDR2 Chipkill FBDIMM Memory Kit	22	507,94 €	11.174,68 €
Express IBM 73.4GB 10K SFF SAS HDD	11	143,67 €	1.580,37 €
Concurrrent KVM Feature Card (StFF) for IBM BladeCenter	11	86,60 €	952,60 €
QLogic 4Gb SFF Fibre Channel Expansion Card for IBM eServer BladeCenter	11	426,19 €	4.688,09 €
Cabina DS4200			
Express DS4200 Model 7V	1	4.692,33 €	4.692,33 €
Express DS4200 500GB 7.2K SATA EV-DDM HDD	11	681,16 €	7.492,76 €

Express DS4200 2-8 Stg. Part.- Fld	1	3.904,13 €	3.904,13 €
Express DS4200 Linux/Intel Host Kit	1	710,04 €	710,04 €
5m Fiber Optic Cable LC-LC	4	90,48 €	361,92 €
Serveis			
3 Year Onsite Repair 9x5x4 Hour Response	1	1.447,49 €	1.447,49 €
Serveis d'instal·lació			1.850,00 €
Total:			**92.050,55 €**

Aquests són els pressupostos de les diferents ampliacions del sistema que hem realitzat posteriorment:

Primera ampliació de disc

Concepte	Qtt	Preu unitari	Subtotal
Express DS4200 500GB 7.2K SATA EV-DDM HDD	5	375,00 €	1.875,00 €
Total:			**2.175,00 €**

Segona ampliació de disc

Concepte	Qtt	Preu unitari	Subtotal
Express IBM System Storage DS4000 EXP420 Storage Expansion Unit	1	3.860,00 €	3.860,00 €
SW 4 Gbps SFP transceiver pair	1	925,00 €	925,00 €
1m Fiber Optic Cable LC-LC	2	29,00 €	58,00 €
Express DS4200 EXP420 Attach 1-3	1	2.785,00 €	2.785,00 €
Express DS4200 1TB 7.2K SATA EV-DDM HDD	3	589,00 €	1.767,00 €
Total:			**10.898,20 €**

Tercera ampliació de disc

Concepte	Qtt	Preu unitari	Subtotal
Express DS4200 1TB 7.2K SATA EV-DDM HDD	4	589,00 €	2.356,00 €
Total:			**2.732,96 €**

Quarta ampliació de disc

Concepte	Qtt	Preu unitari	Subtotal
Express DS4200 1TB 7.2K SATA EV-DDM HDD	9	589,00 €	5.301,00 €
Total:			**6.149,16 €**

Ampliació de memòria

Concepte	Qtt	Preu unitari	Subtotal
8GB (2x4GB) PC2-5300 CL5 ECC DDR2 Chipkill FBDIMM Memory Kit	16	359,50 €	5.752,00 €
Total:			**6.672,32 €**

Annex II, instal·lació del hardware

A continuació descriurem de forma detallada tot el procés seguit per a realitzar la instal·lació del hardware del sistema:

Instal·lació del xassís

El primer pas que hem de fer per a instal·lar el clúster és preparar l'armari on el col·locarem. Disposem d'un rack de 19 polzades IBM el qual disposa de 17 U's lliures (les unitats en que es mesuren els racks, 1U = 1,75” = 44,45mm). Tenim espai suficient per a instal·lar-hi tant el xassís BladeCenter H (9 U's) com la nova cabina de discs DS4200 (3 U's), de tal manera que un cop acabem la instal·lació l'armari quedarà distribuït tal i com podem veure a la *figura 7.1*.

Un cop ens arriba el xassís i tots els mòduls, procedim a instal·lar el BladeCenter. Per motius de seguretat, extraiem tots els components preinstal·lats del xassís per tal d'alleugerir-ne el pes, l'instal·lem al rack, i procedim a connectar-hi tots els mòduls:

- **Power Modules:**

Mòduls que subministren energia elèctrica a tots els elements connectats al xassís. El xassís disposa de 4 slots per a aquests mòduls de subministrament elèctric. Els slots 1 i 2 subministren electricitat als slots de blade 1-7, i als slots I/O 1-4 i 7-10. Els slots 3 i 4 subministren electricitat als slots de blade 8-14 i als slots I/O 5-10. Cada parella de slots subministra corrent a tots els elements de la seva 'zona' de manera redundant: si una font cau, l'altra és capaç d'assumir-ne les seves funcions però s'ha de substituir la font espatllada, ja que no existeix redundància entre les parelles 1-2 i 3-4. Addicionalment a aquesta redundància, el xassís té dues preses de corrent que es connectaran a dues línies de subministrament separades per tal de maximitzar la redundància en aquest apartat.

Inicialment venen instal·lats els mòduls 1 i 2, però, com hi connectarem més de 7 blades, necessitarem instal·lar-hi dos mòduls addicionals (3 i 4).

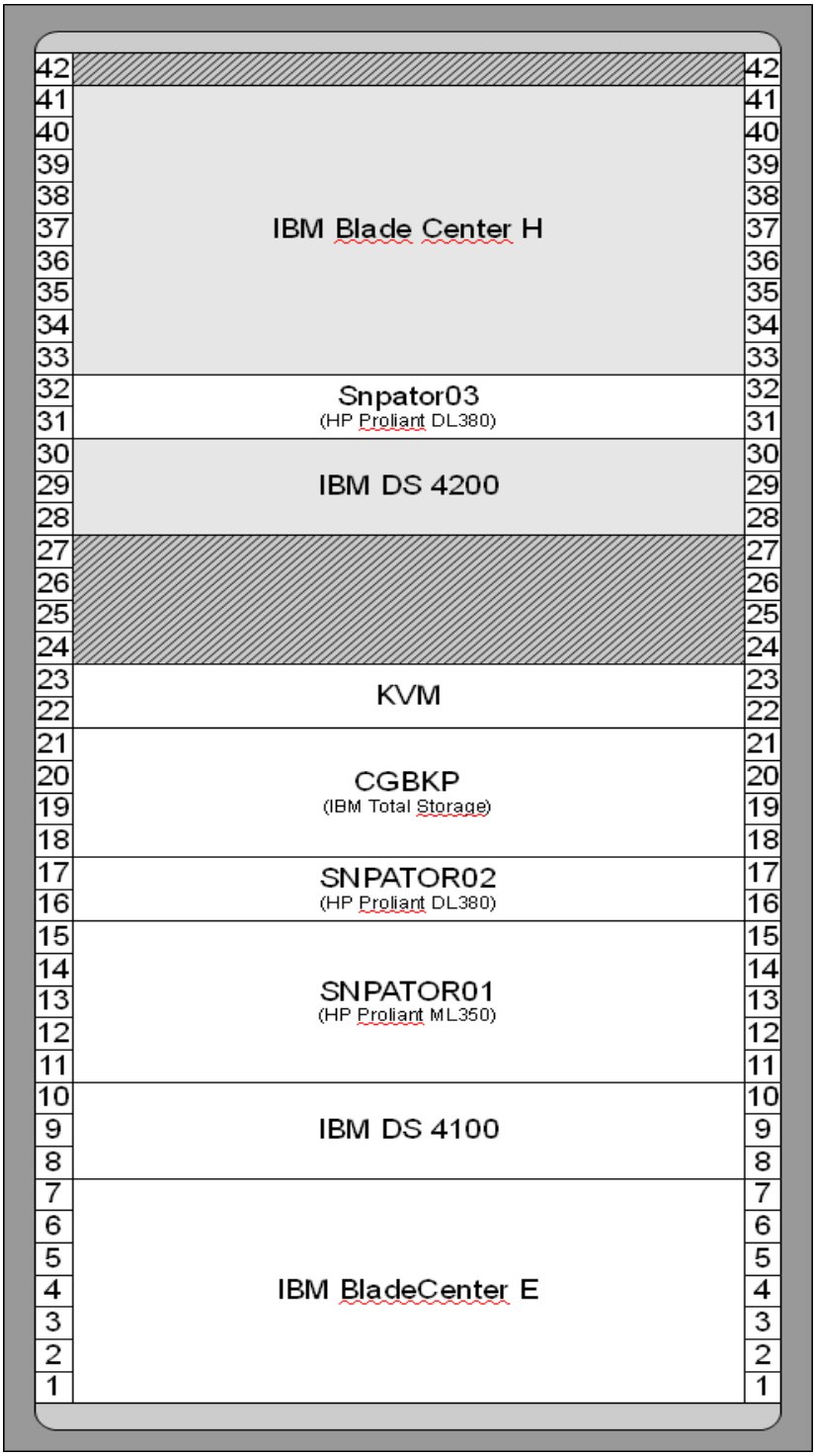

Figura 7.1 – Distribució física dels elements del clúster dins del rack del departament.

- **Media Tray Module:**

Mòdul que inclou dues connexions USB i un lector òptic CD/DVD que permet ser utilitzat per tots els blades. Ve instal·lat per defecte.

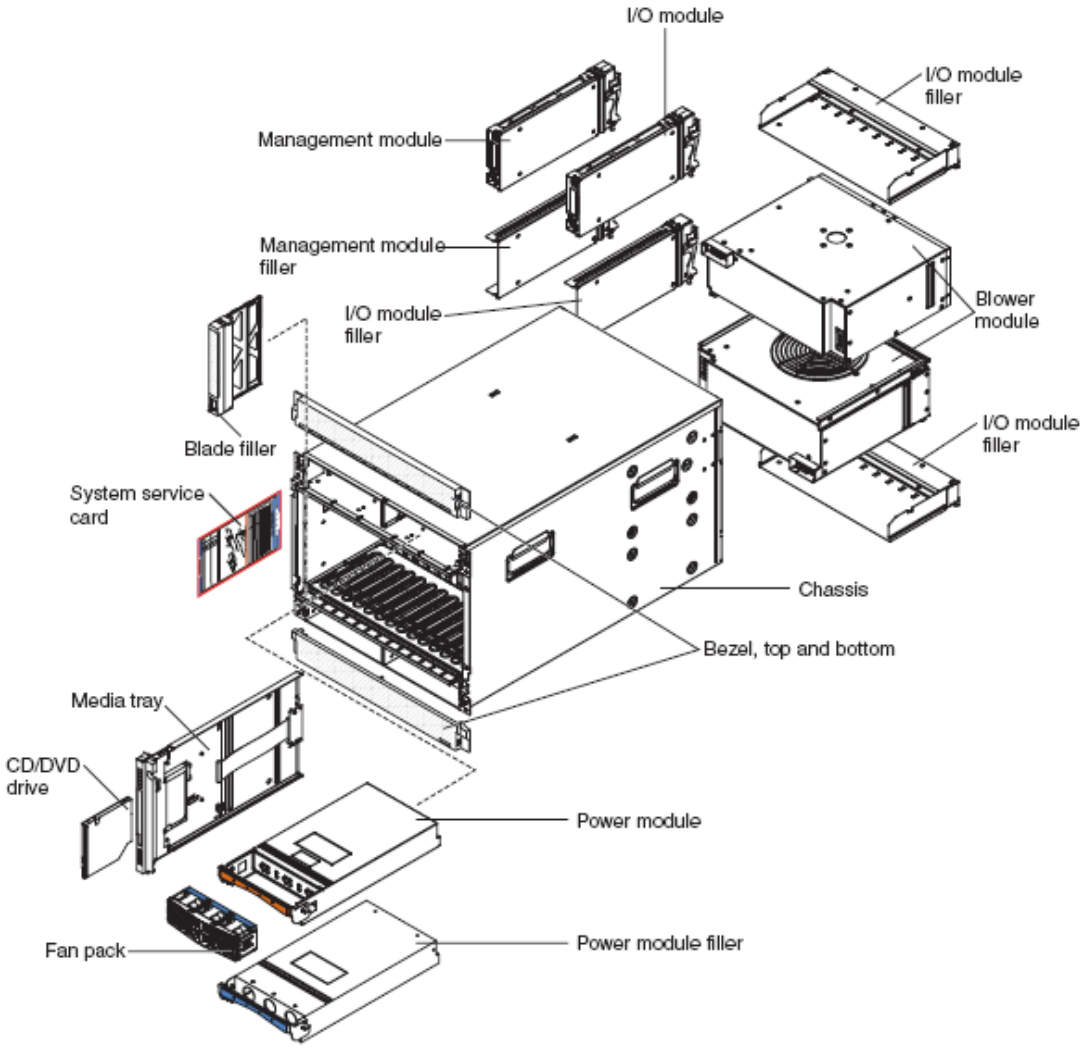

Figura 7.2 - Diagrama de la distribució dels mòduls al xassís.

- **Advanced Management Module:**

Mòdul que permet configurar i administrar tots els components del BladeCenter, així com permet la connexió a dispositius KVM (Keyboard-Video-Mouse), i un connector ethernet per a servir la pàgina web d'administració. El sistema ve de fàbrica amb 1 mòdul hot-swap instal·lat, i permet instal·lar-ne un segon per a redundància, però hem decidit no comprar-lo.

- **I/O Modules:**

10 slots que permeten connectar-hi diversos mòduls d'entrada / sortida, principalment switchs de tot tipus (ethernet, FC, infiniband, ...) i altres dispositius de connectivitat. Inicialment no ve cap instal·lat, però hi instal·larem 2 switchs ethernet 1Gbps i 2 switchs més de Fibre Channel.

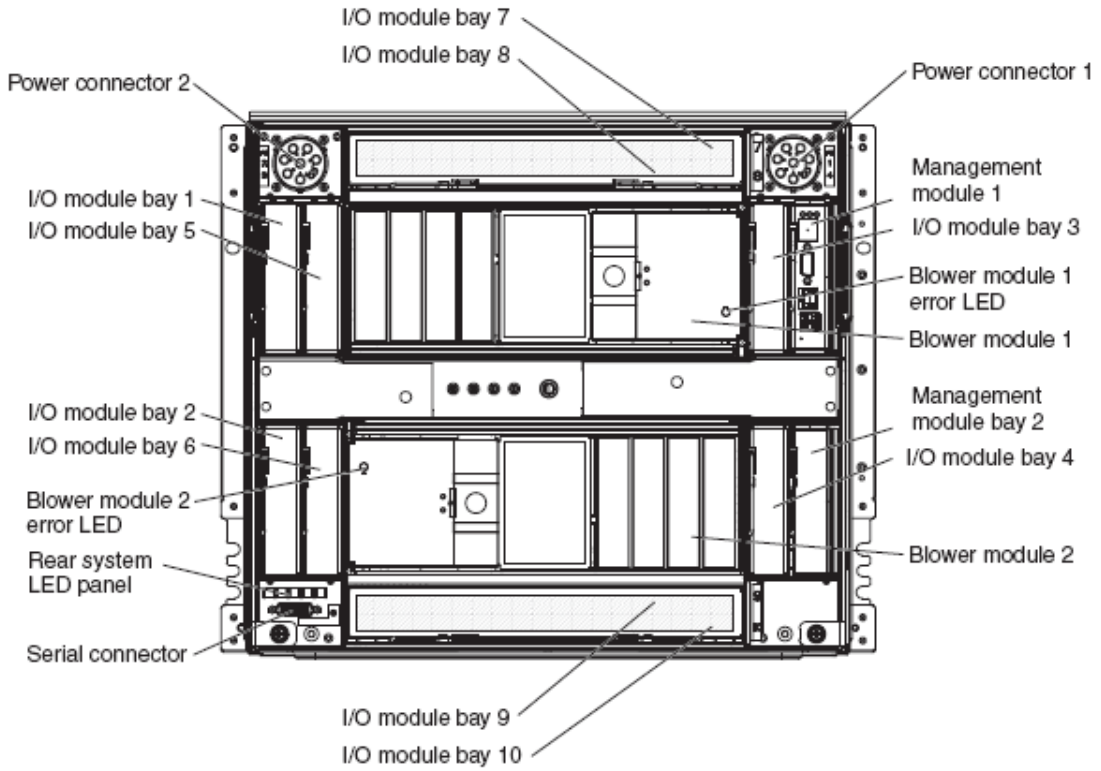

Figura 7.3 - Diagrama posterior dels mòduls del xassís.

- **Blower Modules:**

Mòduls que contenen cadascun d'ells un ventilador de velocitat auto-regulable per a refrigerar tot el sistema. Són redundants i poden ser extrets del sistema per a reduir el pes i facilitar-ne la instal·lació. El sistema porta tots dos instal·lats per defecte.

- **Blade Server Modules:**

Mòduls individuals que conten un servidor complet. Inicialment no en ve cap preinstal·lat, però nosaltres n'instal·larem 11.

Instal·lació dels Switchs Ethernet

Connectem els dos switchs Gigabit ethernet al xassís del BladeCenter. Els switch ethernet "bàsics" com els que hem adquirit s'han de connectar als slots 1 i 2 d'entrada/sortida que trobem a la part del darrere del clúster.

Els switchs no són redundants: cadascun d'ells es connecta només a un dels quatre busos d'intercomunicació del *backbone* del clúster (*figura 7.4*) que porta de sèrie (cada blade té dues interfícies ethernet, però és possible configurar-ne fins a quatre si es compren els components necessaris), de manera que, en cas de caiguda d'un dels switchs, la interfície corresponent a tots els blades quedaria aïllada, no tan sols de l'exterior, sinó també dins del propi xassís.

Aquests switchs disposen també d'una interfície interna de connexió per a comunicar-se amb els management modules, a través de la qual es fan accessibles des del gestor del BladeCenter i de la xarxa.

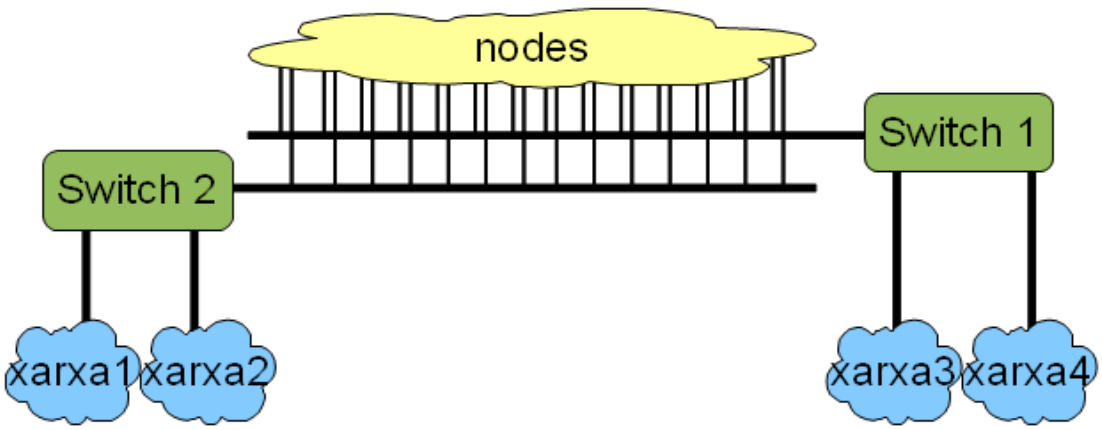

Figura 7.4 - Esquema de connexió al backbone

Instal·lació dels Switchs Fibre Channel

Connectem els dos switchs 4Gb Fibre Channel al xassís del BladeCenter. Els switchs FC "bàsics" s'han de connectar als slots 3 i 4 d'entrada/sortida que trobem a la part de darrere del xassís.

Aquests switchs no tampoc són redundants: cadascun d'ells es connecta a un dels quatre busos de comunicació independents que es troben al *backbone* del xassís, de manera que els blades que disposen de tarja de connexió Fibre Channel es connecten a tots dos busos, i des de cadascun d'aquests busos, arriben a connectar-se al switch. Però, tot i no haver-hi redundància per hardware, sí que és possible crear una redundància efectiva per software utilitzant multipath.

Instal·lació dels Servidors Blade

Tots els servidors blade (*figura 7.5*) venen configurats de fàbrica amb les prestacions inicials, i es necessari completar el muntatge de tots ells amb els components addicionals que hem comprat abans d'instal·lar-los. La configuració inicial de cadascun dels blades és la següent:

- Placa base
- una cpu (Intel Xeon E5345)
- dos dissipadors de calor (un per a cada socket de CPU)
- dos mòduls de memòria de 512 MB cadascun.

Per tal de completar las configuració de cada blade, haurem d'instal·lar la resta dels components a cadascun d'ells:

- una segona cpu (idèntica a l'anterior)
- 4 mòduls de memòria de 2GB cadascun (9GB en total)
- un disc dur SAS de 73,4GB a 10k rpm
- una tarja d'expansió KVM
- una tarja d'expansió Fibre Channel 4Gb (només als bhsrv1 i bhsrv2, els dos blades que requeriran de connexió directa als discs)

Un cop completat el muntatge de tots els blades, els instal·lem dins del xassís, repartint-los per tal de distribuir la càrrega entre les dues parelles de mòduls d'alimentació, especialment separant els servidors que formaran el clúster d'alta disponibilitat, per a assegurar-ne el servei (*figura 7.6*):

- slots 1-7:

  bhsrv1, bhfront, nodes 0 a 3

- slots 8-14:

  bhsrv2, nodes 4 a 7

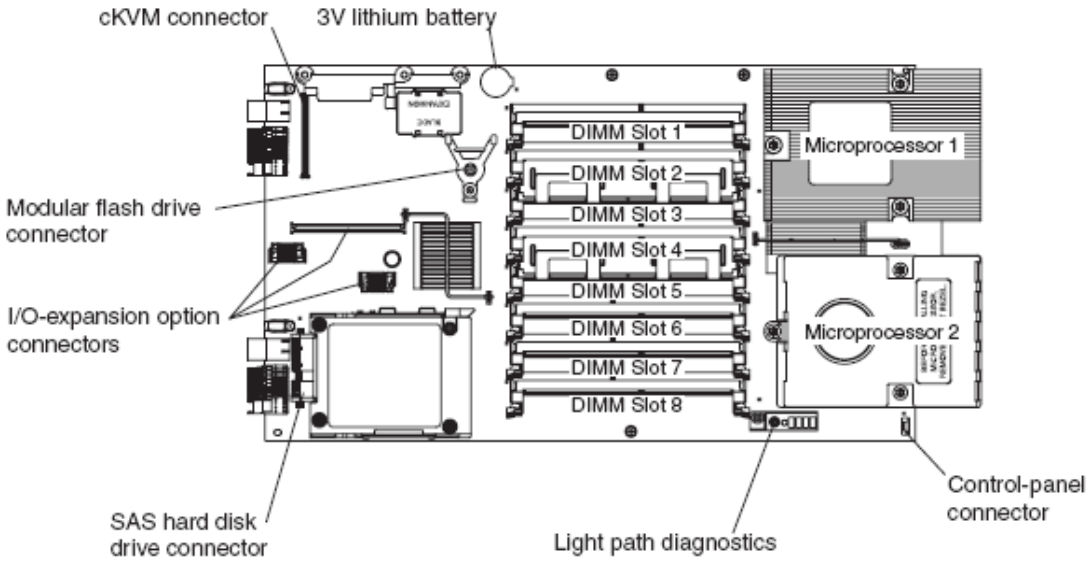

Figura 7.5 – Esquema de components d'un blade IBM HS21.

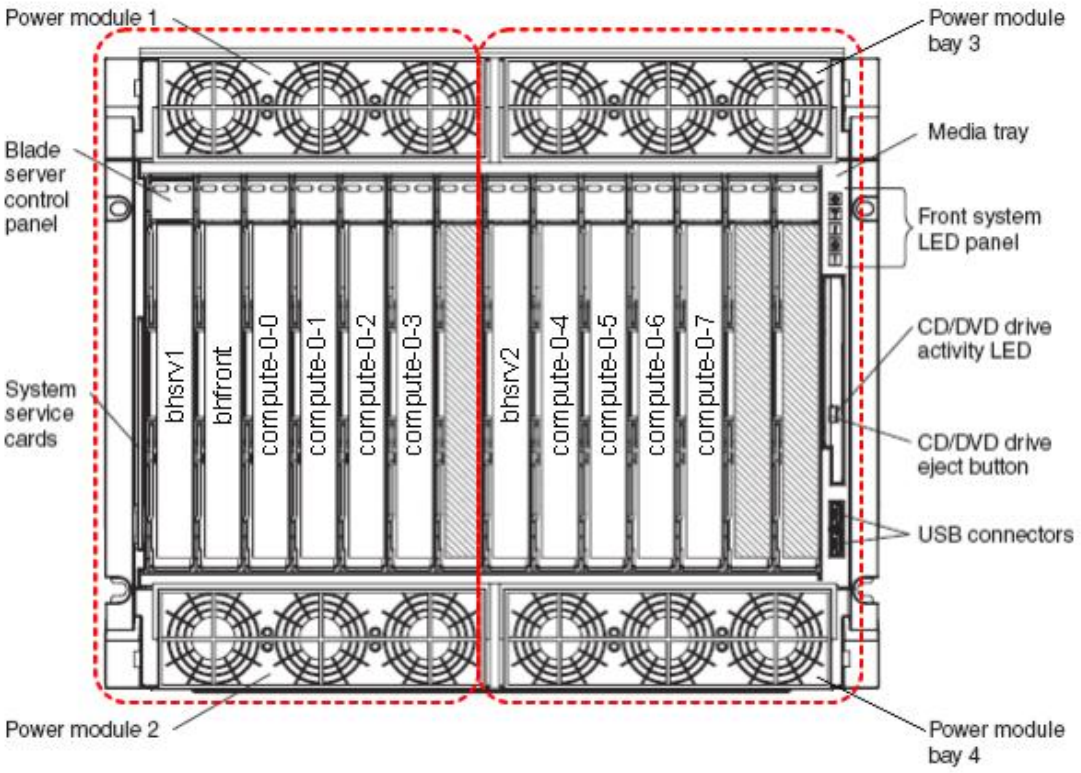

Figura 7.6 - Distribució dels blades segons alimentació

Instal·lació de la cabina

Instal·lem la cabina de disc IBM DS4200 a l'espai de 3U que hem reservat per a ella al rack. La cabina disposa de dues controladores de disc amb connexió Fibre Channel, i dos mòduls d'alimentació elèctrica i ventilació. Aquests mòduls de d'alimentació són plenament redundants i els connectarem a dues xarxes elèctriques diferents. Cadascun dels dos mòduls disposa d'una font d'alimentació capaç de mantenir a tota la cabina, així com dos ventiladors auto-regulables, cadascun dels quals és capaç de substituir a tots els altres durant curts períodes de temps.

Per altra banda, totes dues controladores de disc ofereixen dos ports per a connexions externes (*figura 7.7*), i dos ports més per a connexions a safates d'expansió de discs que seran servides per la pròpia controladora. Cadascuna d'aquestes controladores té control complet sobre tota la cabina, de tal manera que ens permeten tenir redundància sempre i quan realitzem les connexions i la configuració de les màquines de la manera adequada. Addicionalment, cadascuna de les controladores disposa d'una bateria de liti que li permet mantenir la cache de dades fins a tres dies en cas de caiguda de les dues fonts d'alimentació.

Pel que fa als discs, la cabina presenta 16 slots per a discs SATA hot-swap (*figura 7.8*). No hi ha restriccions d'alimentació (balanceig de càrrega), ni a l'hora de crear volums lògics segons el slot que ocupin els discs, de tal manera que col·locarem els 11 discs de que disposem en els slots 1 a 11.

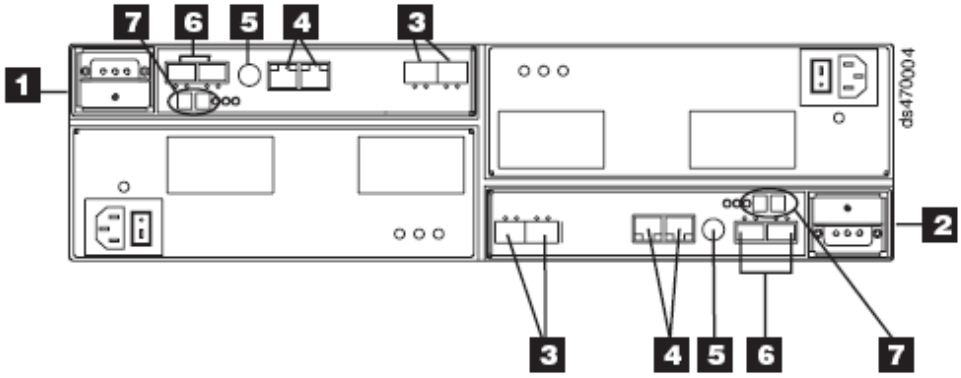

Figure 2. Back view; connectors, switch, ports, enclosure ID for the DS4200 Express

Table 2. Description of Figure 2

Number	Description
1	Controller A
2	Controller B
3	Host channels
4	Ethernet ports
5	Serial port
6	Dual-ported drive channel
7	Enclosure ID

Figura 7.7 – Esquema posterior de les connexions de la cabina de disc IBM DS4200

Figura 7.8 – Esquema frontal de la cabina de disc IBM DS4200

Configuració del xassís

Un cop completat el muntatge tant del xassís com de la cabina, passem a configurar-ne els mòduls de comunicació que hi hem connectat:

Management Module

El Management Module és el dispositiu encarregat de gestionar el xassís així com tots els mòduls que hi tingui connectats. Degut a que aquest actuarà com a punt d'accés a tota la gestió del xassís, és el primer que caldrà configurar.

El Management Module disposa d'una interfície d'administració i gestió via web. Per a accedir-hi cal que ens connectem físicament al port ethernet de que disposa el mòdul i hi accedim a través de la IP i usuari per defecte. Un cop dins l'administrador tenim una gran varietat d'opcions tant de configuració com de control del xassís, però inicialment ens centrarem només en la configuració de la connectivitat del xassís. La configuració que establirem és la següent:

Configuració de xarxa:

El Management Module respondrà al hostname "bhadm" i li assignarem la IP "172.22.203.2" pertanyent a la xarxa d'administració de servidors, de tal manera que, un cop realitzades les connexions, sigui accessible remotament sense necessitat d'accedir físicament a la sala de servidors a través de la URL "http://bhadm.x.upf.edu". Assignem des d'aquí també la configuració de xarxa dels quatre switchs, tant de fibra com ethernet, assignant-los-hi IPs del rang "172.22.203.x". Assignarem també als servidors DNS de la universitat les URL bhswh, bhsw2, bhfc i bhfc2.x.upf.edu per a facilitar-ne l'accés remot.

Configuració d'usuaris:

Eliminem l'usuari per defecte i creem dos usuaris nous, un per als administradors del departament, i un altre per al departament de suport a la recerca de la universitat. El sistema ens permet configurar els usuaris via LDAP, però desestimem aquest sistema, ja

que no preveiem augmentar el nombre d'usuaris que hi han d'accedir, i el LDAP el destinarem només als usuaris del clúster. Configurem el sistema també per a que enviï correus d'alerta en cas que succeeixi qualsevol esdeveniment.

Configuració general i de protocols de xarxa:
Configurem diversos elements del sistema per a adaptar-lo a la nostre xarxa i als seus protocols: NTP, SNMP, DNS i SMTP. El xassís disposa també d'una xarxa interna (separada dels switchs) per a comunicació entre el Management Module i els blades. L'objectiu principal d'aquesta xarxa és l'ús de protocols com LDAP, però com de moment no creiem necessari utilitzar-ho, la deshabilitem. Finalment, actualitzem, des de la pròpia interfície, el firmware del Management Module a la última versió disponible.

Switchs Ethernet

Accedim als switchs ethernet a través de la IP per defecte i en configurem tots els paràmetres per a fer-los operatius: IPs (dins de la subxarxa 203), usuaris, NTP per a mantenir actualitzada l'hora i SNMP per a la gestió de l'ús de la xarxa per part del departament de comunicacions. Un cop tenim configurats els paràmetres bàsics del switch procedim a configurar-hi les interfícies de xarxa. Abans, però, cal que determinem quina serà la topologia de xarxa que farem servir.

Les connexions dels diferents blades quedaran configurades de la següent manera: Tots els blades estan connectats als dos switchs ethernet mitjançant interfícies separades, la eth0 connecta al switch1 i la eth1 al switch2, tot això utilitzant les interfícies internes dels switchs. Tenint en compte aquestes restriccions, hem de complir que les màquines es connectin a les xarxes que necessiten:

- Bhsrv1 i bhsrv2:
  externa (193.145.57.x) i privada per servidors i no enrutable (172.22.201.x).
- Bhfront:
  privada comunicació nodes Rocks-cluster (172.22.205.x) i privada accessible des de la xarxa d'usuaris (172.22.202.x).

- Nodes de càlcul:
  privada no enrutable de servidors per a comunicació amb els servidors NFS (172.22.201.x) i privada de comunicació nodes Rocks-cluster (172.22.205.x).
- Management Module del xassís:
  xarxa de "hardware" i switchs (172.22.203.x) necessària per a l'accés web al software de gestió del BladeCenter i els switchs.

Addicionalment, tenim que els bhsrv1 i bhsrv2 seran els servidors de disc NFS, i es comunicaran amb els nodes de càlcul a través de la xarxa 201. Per altra banda, les connexions entre els nodes de càlcul i el front-end es realitzaran a través de la xarxa 202. Degut a això, ens interessarà que la comunicació a l'interior d'aquestes dues xarxes sigui el més ràpida possible.

Sabent tot això, i que la xarxa 205 no cal que tingui connexió cap a la resta de xarxes de la universitat, veiem que les xarxes que han de ser connectades al switch de la planta de servidors són: 193.145.57.x, 172.22.201.x i 172.22.202.x des dels switchs, i 172.22.203.x des del Management Module, quedant l'estructura de la següent manera (*figura 7.9*):

Un cop tenim definida aquesta estructura, cal que assignem els ports dels switchs a la VLAN corresponent formant l'estructura de xarxa abans descrita:
Switch #1

- VLAN #27 (correspon a la xarxa 193.145.57.x):
  - bhsrv1, bhsrv2, port extern #6
- VLAN #29 (correspon a la xarxa 172.22.205.x):
  - bhfront, nodes de càlcul, port extern #4
- VLAN #4095 (corresponent a la xarxa 172.22.203.x):
  - port de connexió al Management Module 1&2

Switch #2

- VLAN #28 (correspon a la xarxa 172.22.201.x):
  - bhsrv1, bhsrv2, nodes de càlcul, port extern #5 i #6
- VLAN #29 (correspon a la xarxa 172.22.202.x):

- bhfront, port extern #1

- VLAN #4095 (corresponent a la xarxa 172.22.203.x):
  - port de connexió al Management Module 1&2

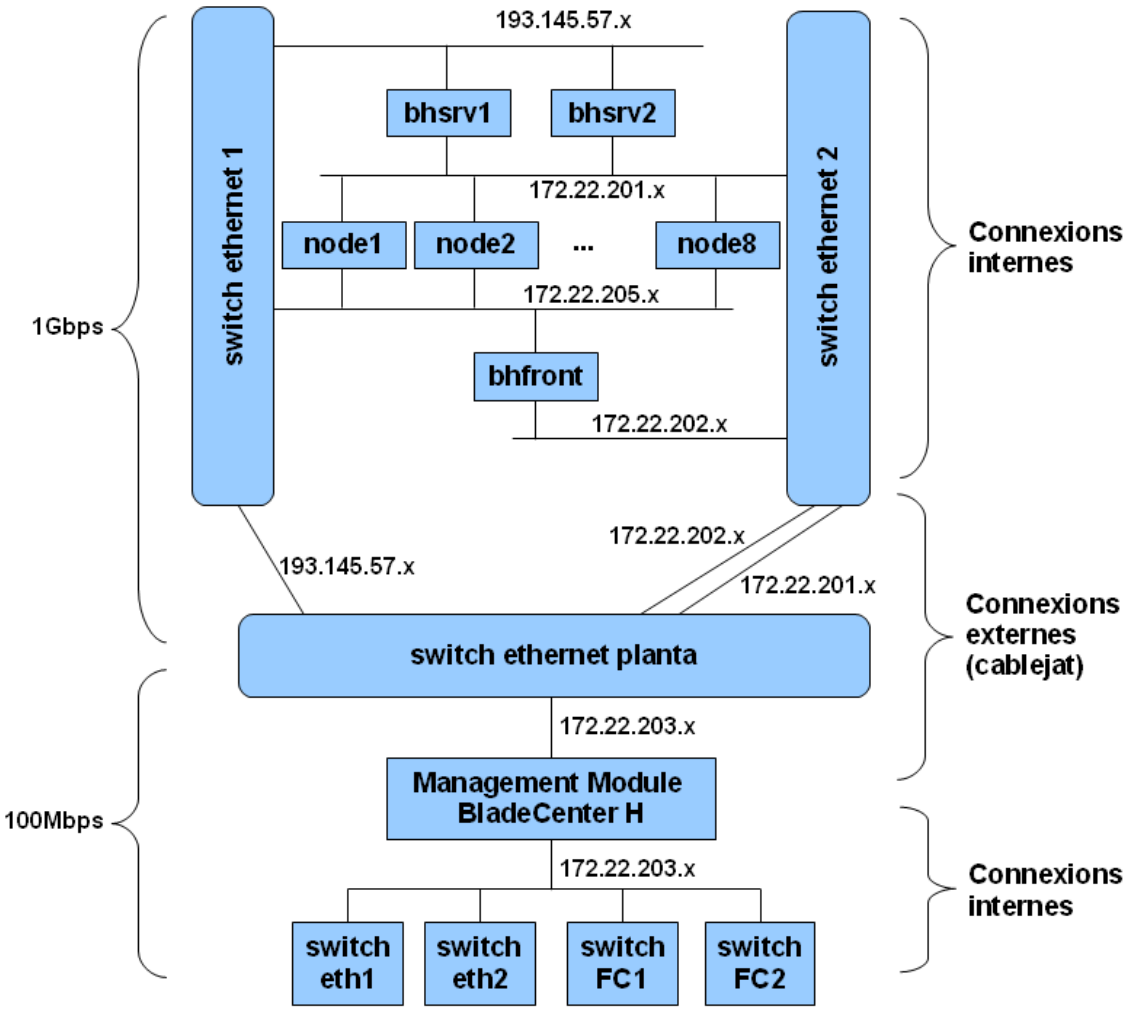

Figura 7.9 - Esquema de la xarxa ethernet de tot el clúster.

Un cop configurada la distribució dels ports dels switch, cal que completem la connectivitat amb la resta de la xarxa. Per a això només caldrà connectar 4 cables ethernet des dels switchs del xassís al switch de planta: un des del port #6 del switch 1 (connectivitat de la xarxa externa), dos des dels ports #1 i #6 del switch 2 (connectivitat de les xarxes 202 i 201 respectivament), i un altre des del port del Management Module (connectivitat xarxa 203). Addicionalment, hem configurat el port #4 del switch 1 i el

port #5 del switch 2 per si és necessari connectar un portàtil a la xarxa 201 o 205 en algun moment, ja que aquestes xarxes no són accessibles des de la resta de xarxes de la universitat.

Switchs Fibre Channel

La xarxa de fibra que tenim actualment està composada per tres servidors (snpator01, 02 i 03) connectats a una única cabina de disc (IBM DS4100) amb dues interfícies de xarxa. Amb l'arribada de la nova cabina i el xassís de blades, caldrà modificar aquesta topologia per tal de connectar els nous elements, tot i mantenir les connexions que teníem fins ara. D'aquesta manera, passaríem de tenir una topologia senzilla (*figura 7.10*), a una topologia força més complexa, ja que necessitarem connectar la nova cabina (DS4200), la qual disposa de dues controladores, i també els blades que seran utilitzats com a nodes del clúster d'alta disponibilitat, els quals muntaran el disc via Fibre Channel i el serviran a la resta de hosts via NFS. Podríem aïllar els dos conjunts de servidors/discs, mantenint l'estructura que ja tenim muntada per una banda, i connectant directament els blades a la nova cabina per l'altre, però d'aquesta manera tindríem les dues "xarxes" aïllades i no seria possible, per exemple, afegir un nou blade que actuï com a servidor (aliè al clúster) i que necessiti accedir a dades contingudes a la cabina antiga. Per tal de possibilitar aquesta i altres futures combinacions, caldrà configurar una topologia més complexa (*figura 7.11*).

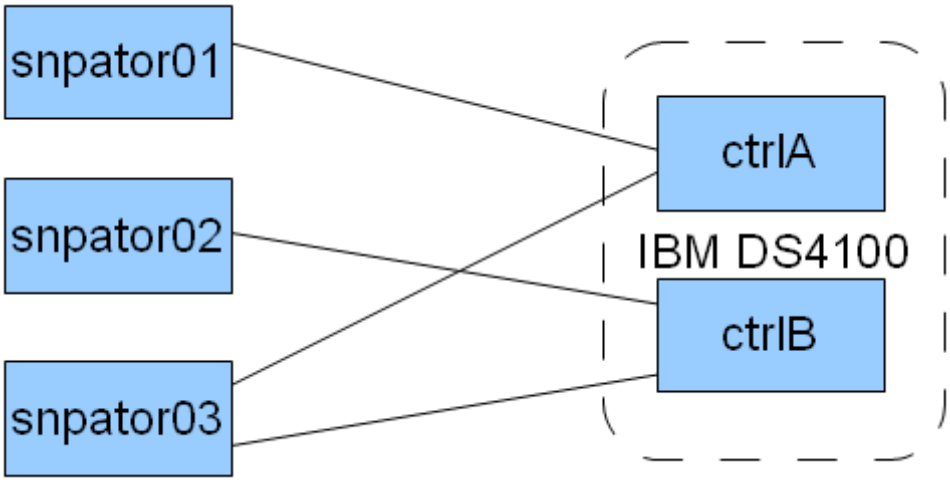

Figura 7.10 - Topologia inicial de Fibre Channel

Un cop definida aquesta topologia, caldrà cablejar el sistema. Per a això, connectarem els SFPs (adaptadors del port del switch al cable de fibra) als ports del switchs que utilitzarem (només pels externs), i connectarem els cables adequadament.

Finalment, accedirem als gestors de configuració de tots dos switchs de fibra a través del navegador, i des d'allà, un cop configurat convenientment el switch, habilitarem les llicències dels ports, ja que per a cada switch tenim 20 ports (14 interns + 6 externs), però només hem els hem llicenciat per a 10. Des d'aquí habilitarem els ports que estem utilitzant (2 interns i 6 externs per a cada switch), i comprovarem la connectivitat de les connexions.

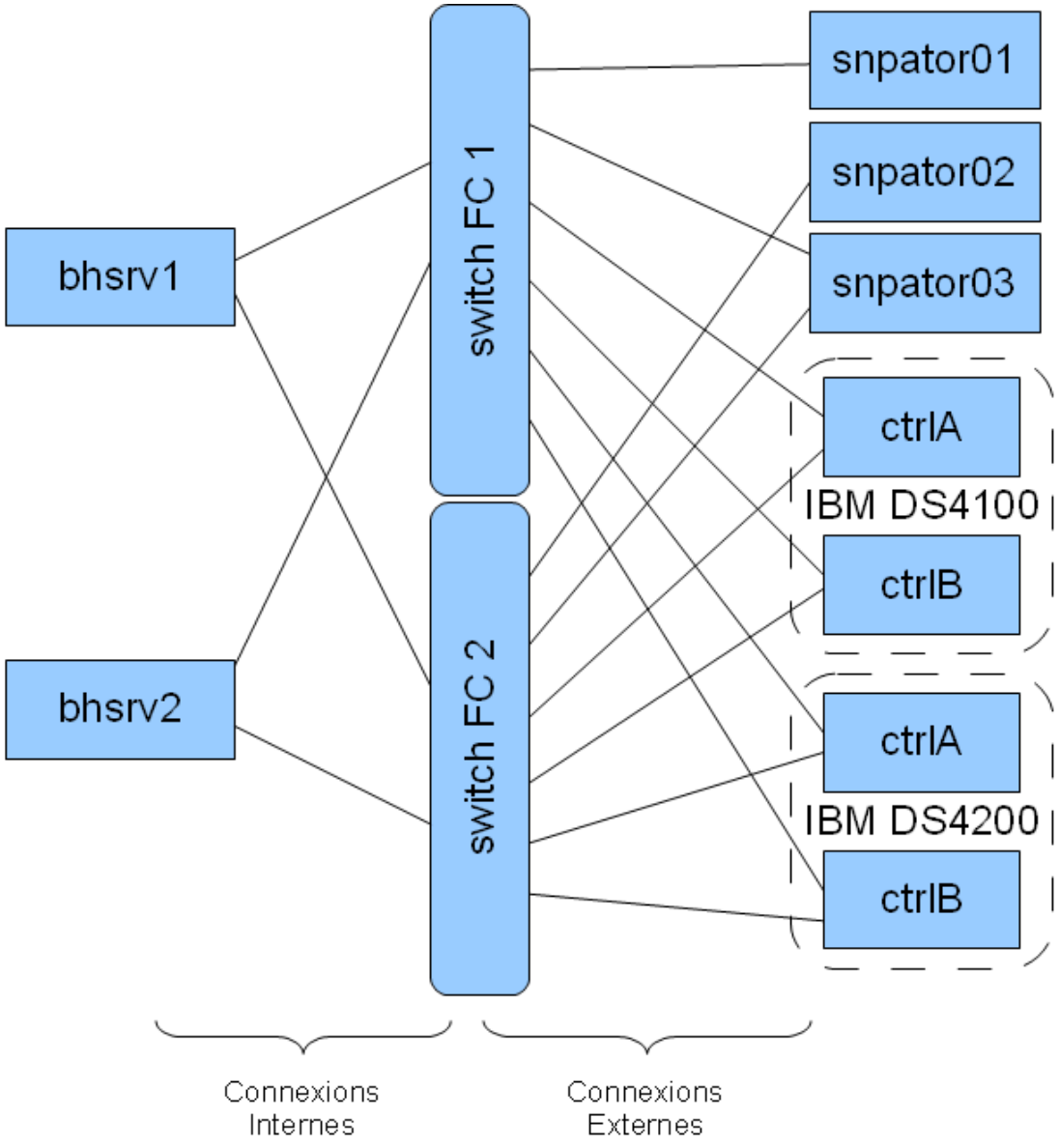

Figura 7.11 - Topologia final de Fibre Channel

Configuració blades

Un cop hem configurat tant els switchs de fibra com els ethernet, cal que comprovem l'últim element del xassís: els blades.

Des del gestor del Management Module podem accedir a l'administrador de blades. Aquí podrem veure tots els blades, així com tot el hardware addicional que tinguin connectat, com les targes KVM que els hem instal·lat a tots, o les targes Fibre Channel que hem instal·lat als que seran bhsrv1 i 2. Un cop comprovem que tots els elements han sigut correctament detectats, descarreguem de la web d'IBM les últimes versions de firmware i drivers, tant dels blades com de les targes FC i KVM, i actualitzem el firmware a l'última versió (els drivers ho farem un cop haguem instal·lat el sistema operatiu del blade).

Configuració KVM

Connectem el servidor KVM al Management Module del xassís del BladeCenter. El funcionament del KVM amb els xassís de blades és lleugerament diferent respecte a la resta de hosts connectats: Al connectar un nou host al servidor de KVM, aquest el detecta automàticament i permet connectar-s'hi simplement escollint-lo al menú. En el cas dels blades però, el KVM ens permet connectar-nos al xassís com a conjunt, i el node al que ens connectem es controlarà a través dels botons de selecció de dispositiu que tenen tots els blades a la seva part frontal.

Per a configurar aquest ús, doncs, caldrà que, un cop el KVM ha detectat al xassís com a conjunt, accedir a l'administrador del Management Module del xassís i des d'allà, a l'apartat “KVM Configuration”, habilitar l'ús del mòdul de KVM per tots els nodes.

Configuració de la cabina de disc

Accedim al software de gestió de les cabines de disc IBM i afegim al gestor les dues cabines de disc, la DS4100 que ja teníem al departament, i la nova DS4200, connectades com a "Out-of-band" (connectades a través d'ethernet, In-band permet fer les gestions a través de FC). Un cop accedim a la configuració de la DS4200, ens detecta la cabina amb 11 discs connectats. Des d'aquí distribuirem els discs en arrays segons l'ús que els donarem. Abans, però, caldrà que decidim quina topologia de discs farem servir.

La documentació de la cabina de disc ens diu que permet crear volums de fins a 8 TB utilitzant discs de 500 GB (és a dir, un únic volum format pels 16 discs que pot tenir la cabina). Tot que això ens permet obtenir la màxima capacitat de cada disc, hem de tenir en compte que la probabilitat de pèrdua de dades creix com més discs formin un mateix grup RAID, degut a la fallada simultània de diversos discs. Per altra banda, la documentació també ens diu que el temps mitjà entre fallades (MTBF) dels discs és d'aproximadament 1,000,000 hores. Sabent això i considerant p = 1/MTBF, calculem les característiques de diferents topologies de disc

Distribució discs	Hot-spare	Espai	% ús espai	Prob. pèrdua	Prob. parada
11 RAID 0	0	5,5 TB	100%	p	p
10 RAID 0	1	5 TB	90%	p^2	p^2
10 RAID 1	1	2,5 TB	45%	1/9 p^3	p^2
10 RAID 5	1	4,5 TB	82%	p^3	p^2
9 RAID 5	2	4 TB	72%	p^4	p^3
5 + 6 RAID 5	0	2+2,5 = 4,5 TB	82%	5/11 p^2	p
5 + 5 RAID 5	1	2+2 =4 TB	72%	4/9 p^3	p^2
3 + 3 + 4 RAID 5	1	1+1+1,5 = 3,5 TB	64%	24/90 p^3	p^2
2 + 2 + 2 + 2 + 2 RAID 1	1	0,5 x5 = 2,5TB	45%	1/9 p^3	p^2

D'aquesta taula podem extreure'n les següents conclusions:

- L'ús de discs Hot-Spare, a part d'augmentar-nos la fiabilitat del sistema, ens permet mantenir el sistema en funcionament en cas de que es produeixi la fallada d'un disc.
- Tenint en compte que disposem d'un servei de manteniment d'IBM el qual, en cas de fallada, ens garanteix el subministrament d'un nou disc al següent dia laborable, l'ús de més d'un disc Hot-Spare no està justificat, al menys amb el nombre de discs amb què treballem actualment.
- Qualsevol sistema de seguretat i redundància de dades que utilitzem comporta una pèrdua d'espai de disc.
- Repartir els discs en diferents arrays comporta pèrdues addicionals pel simple fet d'haver de replicar les mesures de redundància per a cada array.
- Però, degut a això, augmenta la fiabilitat del sistema. Addicionalment, en cas de produir-se alguna pèrdua de dades, aquesta pèrdua queda restringida a un únic conjunt de dades.
- L'ús de RAID 0 és molt arriscat.
- L'ús de RAID 1 ofereix la millor redundància de dades, així com la millor velocitat de lectura, però a costa de perdre un 50% del disc.
- L'ús de RAID 5 ofereix la velocitat més lenta de lectura i escriptura, però a canvi ofereix la millor relació redundància/espai perdut.
- RAID 5 té un cost inicial de discs elevat (1 disc per array, independentment de la mida total), però en ampliacions posteriors de l'array ja no cal dedicar recursos addicionals a la redundància, de manera que permet utilitzar el 100% de les ampliacions de discs.

Tenint en compte tot això, decidim dividir els nostres discs en 3 arrays, destinats cadascun als diferents "conjunts" de dades que necessitem: homes, scratch i "altres" (seran poc usats i alguns són massa petits per a destinar-los-hi un disc o array sencer). Per a tots 3 arrays utilitzarem RAID 5, ja que, tot i que el cost inicial de la redundància és molt elevat (perdem 1 disc per cada partició, a més del disc de Hot-Spare) perdent 4 dels 11 discs, però, com la cabina disposa de 5 slots lliures de disc, en cas de necessitar més espai utilitzarem el 100% dels nous discs que ens arribin. Per altra banda, en cas d'ampliació, aquesta configuració ens permetrà també assignar els nous discs repartits

com vulguem entre les diferents particions depenent de les necessitats quan arribi el moment.

Tenint en compte tot això, la configuració final quedaria de la següent manera:

- Array "homes": 3 discs en RAID 5 (~1TB usable)
    - Logical drive: "homes" ~1TB
- Array "scratch": 3 discs en RAID 5 (~1TB usable)
    - Logical drive: "scratch" ~1TB
- Array "data": 4 discs en RAID 5 (~1.5TB usables)
    - Logical drive: "backup_desk" ~800GB
    - Logical drive: "backup_home" ~500GB
    - Logical drive: "mysql" ~150GB
    - Logical drive: "mysql_binlog" ~50GB
- Hot-Spare: 1 disc

Un cop definits els arrays i *logical drives*, hem de definir els grups de hosts que tindran accés als diferents discs. Inicialment només caldrà donar accés als discs d'aquesta cabina al clúster d'alta disponibilitat, i per això, només crearem un grup "bhusers" que contingui els servidors bhsrv1 i 2, i el qual dóna accés des d'aquestes dues màquines a tots els volums abans creats. Un cop definit aquest grup de hosts, assignarem a cada logical drive un identificador LUN (*Logical Unit Number*) per a poder-los distingir des dels hosts al accedir-hi via Fibre Channel.

Finalment, l'últim pas que realitzarem serà el balanceig de la càrrega de les controladores de la cabina, definint quina serà la controladora preferida per a servir cadascun dels volums. Tal i com estan els discs, els distribuirem a entre les controladores intentant distribuir la càrrega entre totes dues. Per això, posarem en controladores separades les parelles homes/scratch, homes/backup_home i mysql/mysql_binlog. Sabent tot això assignarem a la controladora A els volums homes, backup_desk i mysql_binlog, i a la controladora B els volums scratch, backup_home i mysql. Aquesta configuració no és restrictiva, és a dir, els volums encara poden ser

accedits des de les dues controladores, però el sistema intentarà distribuir el trànsit equitativament per a balancejar la càrrega.

Annex III, instal·lació i configuració del clúster d'alta disponibilitat

A continuació descriurem de forma detallada tot el procés seguit per a realitzar la instal·lació i configuració del clúster d'alta disponibilitat, el qual estarà format per dos servidors físicament idèntics i que disposaran tots dos de la mateixa configuració i serveis (excepte el hostname i la IP). Per tant, tota instal·lació i configuració serà replicada idènticament a les dues màquines.

Instal·lació del Sistema Operatiu

Instal·larem a tots dos servidors SuSE Linux Enterprise Server 10.2, ja que a la universitat disposem de diverses llicències sense utilitzar i estem acostumats a treballar amb aquesta distribució. Realitzarem una instal·lació estàndard d'aquesta distribució, però afegint-hi durant la instal·lació els paquets d'alta disponibilitat, servidor de fitxers, gestió de quotes, servidor web i LAMP, i compiladors. A continuació passem a detallem la configuració dels elements principals del sistema:

Configuració xarxa

Configurem el hostname (`bhsrv1` i `bhsrv2`) i les interfícies de xarxa (eth0 connectada a la xarxa externa i eth1 a la xarxa interna de servidors 201), i afegim les noves IPs al servidor DNS de la universitat.

Configuració Sistema

Configurem l'hora del sistema per a sincronitzar-se utilitzant el servidor NTP de la universitat, i activem el daemon ntpd per a mantenir-la actualitzada.

Configuració Disc

Configurarem tant el disc local del sistema com els discs de la cabina SAN que muntarem via fibra òptica i servirem per NFS.

Distribució del disc local

Per defecte la instal·lació de SLES divideix el disc intern de 70GB en 3 particions, `/`, `swap` i `/var`. Nosaltres però, optarem per a modificar-la utilitzant el *Logical Volume Manager* (LVM). LVM és un sistema de Linux que permet gestionar els volums de disc que muntarem de manera independent dels volums físics que tinguem connectats (*figura 8.1*). Funciona de la següent manera:

Definim un *Volume Group* (VG) el qual contindrà a la resta de volums. A aquest VG, hi podem assignar diversos discs o particions de disc (o qualsevol dispositiu de disc al que podem accedir des del directori `/dev`) els quals s'assignaran com a *Physical Volumes* (PV) amb unes determinades característiques, la principal de les quals és el nombre de blocs. D'aquesta manera, ara el VG que hem creat disposa d'una quantitat de blocs de disc igual a la suma de tots els PV que hi té assignats.

Creem dins del VG un o més *Logical Volumes* (LV), als quals els hi assignarem un nom de dispositiu pel qual podran ser accedits pel sistema com si fossin un dispositiu més (apareixeran a `/dev/mapper`). Finalment, assignem a aquests LV el nombre de blocs que volem que disposin, fins a un màxim del nombre total de blocs que té assignat aquest VG (no és necessari utilitzar tots els blocs, podem deixar-ne sense utilitzar per a futures ampliacions).

Aquest sistema permet, entre altres coses, crear volums lògics molt grans a partir de discs físics més petits, dividir un mateix disc físic en diversos volums lògics independents sense necessitat de particionar, i crear volums lògics d'una mida determinada però mantenint-ne una reserva per a poder ampliar-los més tard.

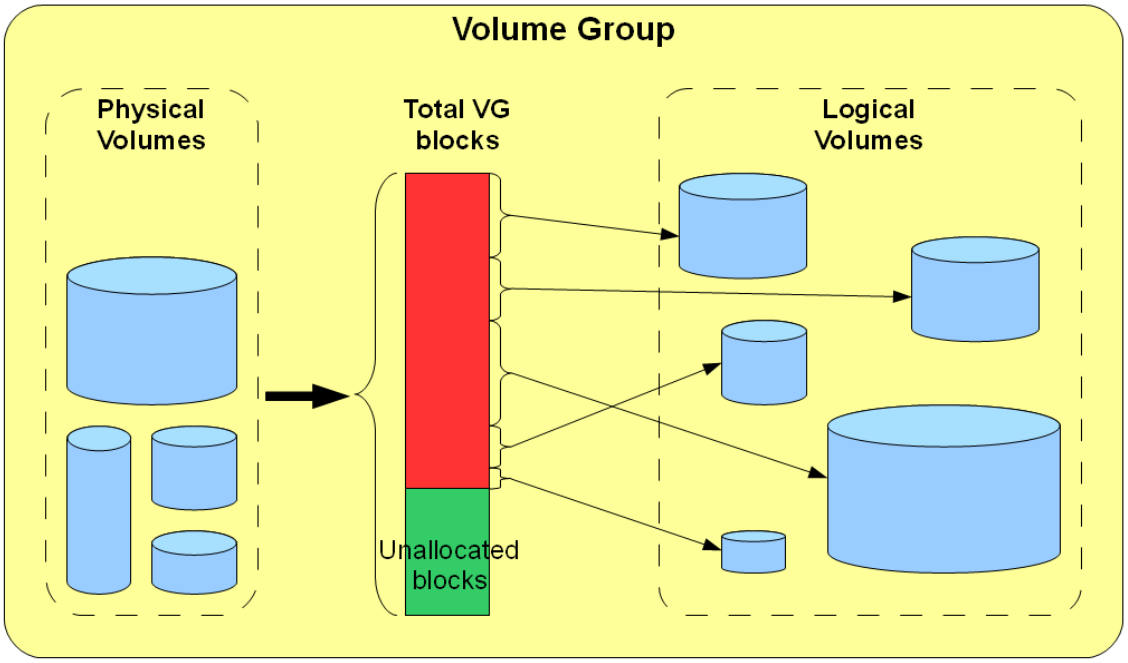

Figura 8.1 - Esquema d'un Volume Group dins de LVM.

Sabent tot això, decidim configurar els disc local de la següent manera durant la instal·lació del sistema operatiu:

- Disc dur local 70GB `sda`.
  - Partició `sda1` de 200 MB assignada a `/boot`.
  - Partició `sda2` de 68,1 GB.
    - Volume Group "`system`":
      - Physical Drives:
        - Partició `sda2` 68,1 GB.
      - Logical Volumes:
        - root (`/`) 20 GB.
        - swap 10 GB.
        - var (`/var`) 10 GB.
        - unallocated 28,1 GB.

Aquestes particions les afegirem a l'arxiu `/etc/fstab` per a ser muntades a l'engegar el sistema utilitzant les opcions "`acl`" i "`user_xattr`" als punts de muntatge `/`, `/boot` i `/var`, mentre que la partició de swap utilitzarà els valors per defecte.

Discs de la Cabina de discs

Pel que fa als discs de la cabina, també utilitzarem Logical Volume Manager, però el procés que haurem de seguir per a connectar-los és lleugerament diferent, ja que la connexió als discs no és directa via el bus intern del servidor, sinó que serà a través de Fibre Channel.

La topologia que hem seguit a l'establir les connexions Fibre Channel entre els servidors i la cabina de discs ens dóna la possibilitat de tenir redundància en les connexions, ja que és possible accedir a qualsevol disc per quatre 4 camins diferents. Aquesta redundància de connexions es veu reflectida en el servidor amb la creació de diversos dispositius de disc, com per exemple:

- `/dev/disk/by-path/pci-0000:06:01.0-fc-0x200500a0b82ac9ab:0x000a000000000000 -> /dev/sdv`
- `/dev/disk/by-path/pci-0000:06:01.1-fc-0x200400a0b82ac9ab:0x000a000000000000 -> /dev/sdw`

que, tot i referenciar realment a un únic disc, ens crea un dispositiu per a cada camí que tenim fins al disc.

Això, com podem veure, ens presenta un problema, i és que, encara que disposem de redundància en les connexions al disc, si muntem el disc utilitzant qualsevol d'aquests dispositius, en cas de fallada d'aquest camí, no podrem accedir al disc per molta redundància que tinguem, ja que aquest dispositiu ens apunta al camí físic del disc enlloc de al disc com a entitat.

Per a solucionar això només caldrà instal·lar el servei "multipath", el qual s'encarrega de detectar tots els camins possibles redundants per a accedir a un mateix disc, crear un

únic dispositiu per a accedir-hi, i gestionar l'accés en cas de caiguda d'algun dels camins. Per exemple, en el cas anterior, abans d'activar el multipath teníem els dispositius:

- `/dev/sdv`
- `/dev/sdw`
- `/dev/disk/by-path/pci-0000:06:01.0-fc-0x200500a0b82ac9ab:0x000a000000000000 -> /dev/sdv`
- `/dev/disk/by-path/pci-0000:06:01.1-fc-0x200400a0b82ac9ab:0x000a000000000000 -> /dev/sdw`
- `/dev/disk/by-path/pci-0000:06:01.2-fc-0x200300a0b82ac9ab:0x000a000000000000 -> /dev/sdx`
- `/dev/disk/by-path/pci-0000:06:01.3-fc-0x200200a0b82ac9ab:0x000a000000000000 -> /dev/sdy`

però a l'activar el servei de multipath se'ns crearan:

- `/dev/dm-10`
- `/dev/disk/by-id/scsi-3600a0b80002ac9a2000018104bd53a1f -> /dev/dm-10`
- `/dev/disk/by-name/ 3600a0b80002ac9a2000018104bd53a1f ->/dev/dm-10`
- `/dev/disk/by-uuid/22b3eaae-a3fd-4579-82e4-0b511858ad18 -> /dev/dm-10`

(on "`3600a0b80002ac9a2000018104bd53a1f`" serà substituït per l'àlies que desitgem si ho configurem així al fitxer `/etc/multipath.conf`).

D'aquesta manera, un cop configurat el multipath, podem muntar el disc utilitzant qualsevol d'aquests nous dispositius "canònics" (que acabin apuntant a `/dev/dm-10`) i conservar la redundància de disc.

Sabent tot això, instal·lem, activem i afegim a la seqüència d'inici el servei `multipathd` i afegim els àlies de tots els volums de la cabina que utilitzarem al fitxer `/etc/multipath.conf`. Un cop configurat el servei, passarem a configurar tots els

discs dins del LVM de la manera més simple, és a dir, per a cada disc individualment, convertir-lo a *Physical Volume*, crear-li un *Volume Group* associat, i creant un *Logical Volume* dins del *Volume Group* utilitzant tot l'espai disponible d'aquell disc. D'aquesta manera obtenim el mateix resultat que si no haguéssim utilitzat el LVM, però això ens permetrà ampliar els discs utilitzant el LVM en un futur (ampliar particions mitjançant LVM és molt més ràpid que assignar les ampliacions a nivell de volums de cabina i aplicar les ampliacions al sistema). Un cops creats tots els Logical Volumes, formatarem totes les particions en ext3, que és el sistema de fitxers que hem escollit.

El següent i últim pas seria habitualment afegir aquestes particions al fitxer `/etc/fstab` per tal que es muntessin a l'engegar el servidor. Les opcions que utilitzaríem per a elles serien:

- homes
    - `acl,user_xattr, usrquota, grpquota`
    - muntat a `/sp/fs/homes`
- scratch
    - `acl,user_xattr`
    - muntat a `/sp/fs/scratch`
- backup_desk
    - `acl,user_xattr, usrquota, grpquota`
    - muntat a `/sp/fs/backup_desk`
- mysql_user
    - `acl,user_xattr`
    - muntat a `/sp/fs/mysql_user`
- mysql_user_binlog
    - `acl,user_xattr`
    - muntat a `/sp/fs/mysql_user_binlog`

Aquestes són les opcions de muntatge de disc que utilitzaríem per a aquests volums, típicament al fitxer fstab, però com utilitzarem el sistema d'alta disponibilitat i els discs podran ser usats indiscriminadament pels dos servidors, no podran ser muntats a l'inici sinó que establirem aquesta configuració de les particions al configurar el servei linuxHA.

Tot i que no muntarem ara aquests volums, sí que muntarem els volums homes i backup_desk, per tal de crear-hi els fitxers de sistema que necessitaran per a poder treballar amb quotes. Per tal de crear-los farem per a cada volum "`quotacheck -vug /mountpoint`", i quan muntem definitivament els filesystems, haurem d'executar `quotaon` (i `quotaoff` per a desmuntar-los).

Cal destacar que el volum backup_homes no apareix aquí ja que, tot i estar a la nostra cabina de disc, serà muntat i gestionat per un altre servidor del departament, el qual l'utilitzarà com a emmagatzemament intermedi abans d'enviar els backups al robot de cintes del grup.

Estructura dels discs

L'estructura de directoris d'aquests discs la distribuirem de la següent manera:

- homes
  - Muntat a `/sp/fs/homes`.
  - A l'arrel del volum hi tindrem els fitxers necessaris pel sistema de quotes (`aquota.users` i `aquota.groups`), el directori de sistema `lost+found`, i un directori "homes", dins del qual hi trobarem:
    - aplic: directori on hi instal·larem tot el software que necessitin els usuaris del clúster de càlcul.
    - rocks: directori d'administració amb scripts de manteniment d'ús comú pels nodes del clúster de càlcul.
    - shared_libs: directori de llibreries compartides pels nodes del clúster de càlcul.
    - users: homes dels usuaris.

- scratch
  - Muntat a `/sp/fs/scratch`.
  - A l'arrel del volum hi trobarem el directori de sistema `lost+found`, i un directori "scratch", dins del qual hi trobarem un directori personal per a cadascun dels usuaris del clúster de càlcul.

- backup_desk
  - Muntat a `/sp/fs/backup_desk`.
  - A l'arrel del volum hi tindrem els fitxers necessaris pel sistema de quotes (`aquota.users` i `aquota.groups`), el directori de sistema `lost+found`, i un directori "`backup_desk`", dins del qual hi trobarem un directori personal per a cadascun dels usuaris del sistema.

- mysql_user
  - Muntat a `/sp/fs/mysql_user`.
  - A l'arrel del volum hi tindrem tres directoris:
    - config: contindrà els fitxers de configuració de mysql.
    - mysql: contindrà les dades i l'estructura de directoris necessària per a la base de dades.
    - scripts: contindrà scripts d'administració de mysql.

- mysql_user_binlog
  - Muntat a `/sp/fs/mysql_user/mysql/bin-log`.
  - Contindrà les dades necessàries pel log binari, no hi crearem cap estructura de directoris especial.

Finalment, per tal de consolidar la estructura de fitxers (de manera que sigui independent de quina màquina estigui servint/tingui muntats els volums en un moment donat), crearem els següents links:

- `/home/homes -> /sp/fs/homes/homes`
- `/home/scratch -> /sp/fs/scratch/scratch`
- `/home/backup_desk -> /sp/fs/backup_desk/backup_desk`
- `/homes -> /home/homes`
- `/scratch -> /home/scratch`
- `/backup_desk -> /home/backup_desk`

Configuració de Serveis

A continuació descriurem la configuració dels diferents serveis que oferirà el clúster d'alta disponibilitat.

Configuració NFS

La configuració del servei NFS per tal de fer accessibles les dades del disc a totes les màquines que formin el clúster es divideix en 3 apartats: la definició i configuració dels volums a exportar, la configuració dels servidors per a fer accessibles aquests volums, i la configuració dels clients per a muntar-los.

Per a definir i configurar la exportació dels volums via NFS ens calen 3 coses: el directori arrel del volum a exportar, les IPs dels clients que hi tindran accés i les opcions de configuració de NFS que seran efectives per aquest volum. Els volums que volem fer accessibles des de tots els servidors que conformin el clúster de càlcul seran "*homes*", "*scratch*" i "*backup_desk*". Com volem servir els volums complets i no només un subdirectori, hem d'indicar el seus punts de muntatge, que són, respectivament, `/sp/fs/homes`, `/sp/fs/scratch` i `/sp/fs/backup_desk`.

Els clients que tindran accés als volums via NFS seran:

- Per la xarxa externa "193.145.57.x"
    - bhsrv1, bhsrv2 i les seves IPs flotants bhusers i bhscratch
- Per la xarxa interna "172.22.201.x"
    - bhsrv1, bhsrv2, les seves IPs flotants bhusers i bhscratch, i tots els servidors que formin part del clúster de càlcul. Per tal de simplificar la configuració i evitar problemes en cas d'ampliar el clúster, donarem accés a tota la xarxa 201
- Per la xarxa interna "172.22.202.x"
    - El front-end del clúster bhfront.

Finalment, el servidor NFS accepta diversos paràmetres de configuració independents per cadascun dels volums que servim. A continuació detallarem aquests paràmetres per tal de veure quins s'adeqüen millor a la configuració del nostre sistema:

- `rw / ro` : accés de lectura-escriptura o només lectura. Donarem accés `rw` a tots els volums per a totes les màquines.
- `sync / async`: El protocol de NFS treballa per efecte de manera síncrona, tal que el servidor només respongui a una petició d'escriptura un cop aquesta ha sigut realitzada sobre el disc. En mode asíncron, en canvi, el servidor respon a les peticions d'escriptura sense consultar si han finalitzat o no. El mètode asíncron suposa una millora en el rendiment d'accés a disc, però a canvi de posar en perill la integritat de les dades en cas d'apagada inesperada. Com no podem córrer aquest risc, escollim el mode síncron.
- `root_squash / no_root_squash`: Aquesta opció, quan està activada, fa que tota petició realitzada al client per l'usuari root, es realitzi al servidor des d'un altre usuari (per defecte "nobody"). Aquesta és una opció de seguretat interessant, i inicialment l'activarem, però potser trobem que és necessari desactivar-la en algun moment durant la configuració i avaluació del sistema gestor de cues.
- `subtree_check / no_subtree_check`: Per defecte, com NFS permet servir tant filesystems sencers com només subdirectoris dins d'ells, quan el servidor rep una petició, ha de comprovar si l'arxiu es troba dins de l'estructura de directoris que pot servir. Com servirem i muntarem les particions completes, desactivem aquesta opció, cosa que també millorarà el rendiment.
- *secure / insecure*: Realitzar la comunicació utilitzant ports per sota del 1024. Per defecte està activat i no tenim cap motiu per a canviar-ho.
- *wdelay / no_wdelay*: Al treballar en mode síncron, el servidor pot decidir esperar-se lleugerament abans de realitzar una operació d'escriptura quan sospita que properament poden arribar més peticions d'escriptura relacionades, millorant-ne el rendiment. Però, per altra banda, si les peticions són aleatòries i aquestes escriptures no arriben, el rendiment empitjora. D'entrada no sabem quin efecte pot tenir aquesta opció, de manera que deixarem l'ús per defecte (activada), i més endavant, durant el procés d'avaluació i benchmarking veurem

si desactivar-la ofereix cap millora.

- `hide / nohide`: La opció `nohide` només té sentit si volguéssim servir filesystems que estiguessin muntats dins d'altres filesystems que també estiguéssim servint. Com no tenim aquesta configuració, utilitzem `hide`.
- `secure_locks / insecure_locks`: La opció `insecure_locks` és útil només per compatibilitat amb implementacions antigues del protocol NFS. Utilitzarem `secure_locks`.
- `acl / no_acl`: La opció `no_acl` és útil per motius de compatibilitat amb implementacions antigues de NFS v2 i v3, les quals gestionen les acl a nivell local. Implementacions més modernes utilitzen el *accesrpc*, de manera que és el propi servidor NFS qui gestiona els accessos. Utilitzarem `acl`.
- `mountpoint`: aquesta opció força al servidor a comprovar si el volum que ha de servir ha estat correctament muntat al servidor abans de poder servir-lo (i evitar així servir filesystems buits, o les dades del directori on s'hauria d'haver muntat el volum). Ja que tots dos servidors de disc formaran un clúster d'alta disponibilitat i caldrà que les particions puguin passar a estar muntades d'un servidor a l'altre, activarem aquesta opció.
- `fsid`: El servidor NFS, a l'utilitzar manegadors o les propietats del fitxers utilitza per defecte com a identificador del filesystem un nombre derivat dels blocs del dispositiu on està muntat el sistema de fitxers. Aquesta opció permet sobrepassar aquest sistema, de manera que fixem un nombre de 32 bits qualsevol, gràcies al qual podem garantir l'estabilitat dels manegadors de fitxers a en cas de "canvi" de servidor al tenir configurat el clúster d'alta disponibilitat. Per a garantir-ne el correcte funcionament, caldrà utilitzar un nombre únic per cada volum que exportem i hauran de ser els mateixos a tots els servidors NFS en que els configurem.

La configuració del servidor NFS es divideix en diversos passos:

- Configuració del daemon nfsd, mitjançant el fitxer `/etc/sysconfig/nfs`. Dins d'aquest fitxer podem configurar paràmetres com:
    - Nombre de threads de servidor NFS que han de córrer al kernel:

      Per defecte 4, idealment un per cada connexió NFS que hagi de suportar

el servidor (és a dir, en el nostre cas 8x8=64). Utilitzar un nombre molt gran pot provocar la saturació del servidor NFS. Inicialment utilitzarem 32, però caldrà testejar aquest nombre durant la fase de benchmarking.

- Port del mountd:
  Per defecte sense establir, però és possible configurar-lo per a poder treballar a través d'un firewall. El deixarem en blanc ja que el portmapper s'encarregarà de gestionar-lo.
- Activar el mòdul de seguretat GSS:
  Només l'activarem si utilitzem realment el protocol NFS v4. Com inicialment farem servir la versió 3, no l'activarem.
- Suport al protocol NFSv4:
  L'activarem, però inicialment utilitzarem NFS v3. Durant la fase de proves veurem quina de les dues versions es mostra més eficient.

- Configuració dels fitxers `/etc/hosts.allow` i `deny` per tal d'establir quines màquines de la xarxa tindran accés als serveis d'aquest servidor.
  - Donarem accés als mateixos servidors que hem donat accés al fitxer `/etc/exports` (bhsrv1&2, bhusers, bhscratch, bhfront i tota la subxarxa 201) a tots els serveis necessaris per a utilitzar el servidor NFS: *portmap* (assignació dinàmica dels ports a través del firewall), *lockd* (control de bloqueig de fitxers), *mountd* (muntatge de sistemes de fitxers remots), *rquotad* (quotes remotes) i *statd* (estat dels fitxers remots).
  - Bloquejarem l'accés a la resta de serveis per a totes les màquines. És important recordar això de cara a l'activació de futurs serveis. Per exemple, per a donar accés al servei sshd, caldrà afegir "`sshd`" al fitxer `/etc/hosts.allow`

- Engegar els daemons necessaris per a utilitzar el servidor NFS. Aquests daemons són *portmap*, *lockd*, *mountd*, *statd*, *rquotad* i *nfsd*. Afortunadament, aquesta distribució de Linux inclou el servei "*nfsserver*", el qual engega automàticament tots els serveis necessaris per al funcionament del servidor NFS excepte el rquotad, el qual haurem d'activar a part degut a que ve instal·lat en un mòdul diferent. Tot i així, com aquest servei dependrà del clúster d'alta disponibilitat, només configurarem el rquotad per a que s'iniciï automàticament a

l'engegar el sistema.

Un cop hem configurat el servidor NFS, ja hem acabat de configurar aquest servidor, ja que no caldrà que muntin de nou les particions que ells mateixos estan servint. Però en cas que volguéssim muntar les particions NFS, només caldria tenir en marxa els serveis portmapper, statd i lockd, i un cop fet això, podem muntar els volums mitjançant el sistema que preferim, podent escollir entre les opcions de muntatge soft (si la crida al fitxer nfs falla, informa al procés i, generalment, acaba amb error) o hard (si la crida falla, el procés es bloqueja fins que el fitxer estigui disponible).

Configuració Ldap

Utilitzarem OpenLDAP com a servidor LDAP. Aquest servei el configurarem de manera diferent a la resta de serveis del clúster d'alta disponibilitat, ja que, degut a la seva importància, enlloc de fer una instal·lació idèntica a tots els nodes del servidor d'alta disponibilitat, el que farem serà configurar-lo a un dels servidors (bhsrv2) com a mestre i a l'altre (bhsrv1) com a esclau, el qual replicarà la informació continguda al mestre. Per tal d'establir aquesta configuració, caldrà realitzar primer una instal·lació estàndard del servei, i, posteriorment, diferenciar-lo a totes dues màquines.

Instal·lació bàsica del servei

Configurem el servidor de LDAP seguint la configuració estàndard. Utilitzarem els esquemes (fitxers que defineixen els atributs de la base de dades LDAP) que suggereix LDAP per defecte: “core”, “cosine”, “inetorgperson”, “rfc2307bis” i “yast”. També necessitarem afegir a la configuració l'esquema “samba3”, per tal de permetre l'accés via SAMBA als usuaris un cop en configurem el servidor. Pel que fa a la resta d'opcions, activarem l'opció per acceptar peticions de tipus “LDAPv2 Bind”, i l'ús del protocol TLS (Transport Layer Security) per tal d'assegurar-ne les comunicacions. Configurem la base de dades pròpiament, definint com a base del nom domini “`dc=upf,dc=edu`” i creant-ne l'usuari root, i finalment, configurem el servei per a que s’iniciï automàticament a l'engegar el sistema.

Configuració del servidor Master

Hem de modificar el fitxer de configuració del servei LDAP al servidor que actuarà com a master, per tal que actuï com a tal. Aquest canvi consisteix en afegir al fitxer `/etc/openldap/slapd.conf` el següent:

- Per cada servidor esclau on hi replicarem la base de dades ldap, una línia definint la uri de l'esclau i com a quin usuari de la base de dades s'hi connectarà:

```
replica host=bhsrv1.upf.edu
     bindmethod=simple
     binddn="cn=Sync_user,dc=upf,dc=edu"
     bindmethod=simple credentials=secret
```

- El directori on el servei SLURPD (l'encarregat de mantenir la consistència entre el master i tots els esclaus) hi desarà els logs

```
replogfile /var/lib/ldap/slurp.log
```

Configuració del servidor esclau

Anàlogament a les modificacions realitzades al master, hem de modificar el fitxer de configuració de LDAP al servidor esclau per tal de que es comporti com a tal. El mateix fitxer `/etc/openldap/slapd.conf` que hem configurat al servidor master, aquí l'haurem de modificar de la següent manera:

- Definir la uri del servidor master, i com a quin usuari es realitzaran les modificacions de la base de dades:

```
updatedn "cn=Sync_user,dc=upf,dc=edu"
updateref ldap://bhsrv2.upf.edu
```

- Donar permisos d'escriptura a tota la base de dades a l'usuari que realitzarà la rèplica des del master:

```
access to *
        by dn.exact="cn=Sync_user,dc=upf,dc=edu" write
        by dn.exact="cn=Sync_user,dc=upf,dc=edu" read
        by dn.exact="cn=Sync_user,dc=upf,dc=edu" auth
        by * none break
```

És important destacar que aquesta entrada ha de ser la primera de l'apartat d'ACLs del fitxer de configuració, ja que les peticions d'accés només tenen en compte la primera entrada amb la que coincideixen, de manera que així ens assegurem que aquest usuari sempre tingui permisos d'escriptura.

- Activació del servei de replicació:

  Un cop configurats els servidors LDAP, ja només cal activar el daemon SLURPD al servidor master i configurar-lo per a que arranqui automàticament a l'engegar el sistema. Abans, però, queda pendent un últim pas: Crear a la base de dades una entrada per a l'usuari que ha de realitzar la còpia de la rèplica. Per a crear-lo, des del servidor master executarem:

```
cat<<EOF | ldapadd -x -h bhsrv2.upf.edu
-D "cn=Administrator,dc=upf,dc=edu" -w passadmin
     > dn: cn=Sync_user,dc=upf,dc=edu
     > cn: Sync_user
     > objectclass: organizationalRole
     > objectclass: simpleSecurityObject
     > userPassword: passuser
     > EOF
```

Un cop creat, només caldrà que realitzem manualment una còpia de la base de dades del master cap a l'esclau (la qual es troba a `/var/lib/ldap/`), ja que, d'altra manera, la replicació no funcionarà perquè aquest usuari no arribaria mai autentificar-se.

- Habilitació del servei:

  L'últim pas a realitzar serà habilitar els serveis ldap i ldaps a la interfície externa del firewall, així com habilitar l'ús del servei slapd al fitxer `hosts.allow` per a

totes les màquines que formaran part del clúster.

Instal·lació del client LDAP

Finalment, configurarem el client LDAP per tal de fer que els servidors del clúster d'alta disponibilitat puguin utilitzar els usuaris continguts a la base de dades LDAP.

- Servidor LDAP: "bhsrv2.upf.edu bhsrv1.upf.edu". D'aquesta manera, el sistema intentarà connectar primer al servidor master, i si aquest falla per qualsevol motiu, intentarà obtenir la informació de l'esclau.
- DN base: "`dc=upf,dc=edu`".
- TLS/SSL: sí.
- Mapping d'usuaris: "`ou=Users,dc=upf,dc=edu`".
- Mapping de passwords: "`ou=Users,dc=upf,dc=edu`"
- Mapping de grups: "`ou=Groups,dc=upf,dc=edu`".
- Protocol pel canvi de passwrd: "`crypt`".

A l'establir aquesta configuració, se'ns dóna la opció d'habilitar el login per a aquests usuaris en aquestes màquines. Idealment no seria necessari, ja que pel funcionament del servidor de disc només caldria veure els usuaris LDAP, però no que aquests usuaris accedissin al sistema. Cal tenir en compte, però que tots els elements del clúster de càlcul estaran dins de la xarxa interna de la universitat, la qual només és accessible des de l'exterior utilitzant una VPN. Com és possible que es donin situacions en què un usuari no pot fer servir aquesta VPN (estància a l'estranger on els ports de la VPN estan bloquejats), habilitarem l'opció de login, però en bloquejarem per defecte l'accés a tots els usuaris (mitjançant `/etc/security/access.conf`) i habilitarem els usuaris un a un segons sigui necessari.

Configuració d'usuaris i grups

Tot i que hem definit el mapping dels usuaris i grups LDAP dins de l'estructura "`ou=Users,dc=upf,dc=edu`" i "`ou=Groups,dc=upf,dc=edu`", aquests elements "ou" no existeixen a l'estructura per defecte de LDAP i caldrà crear-los. Per a això farem servir les següents comandes:

```
# cat <<EOF | ldapadd -x -h bhsrv2.upf.edu -D
     "cn=Administrator,dc=upf,dc=edu" -w password
> dn: ou=Groups,dc=upf,dc=edu
> ou: Groups
> objectclass: top
> objectclass: organizationalUnit
> EOF
```

```
# cat <<EOF | ldapadd -x -h bhsrv2.upf.edu -D
     "cn=Administrator,dc=upf,dc=edu" -w password
> dn: ou=Users,dc=upf,dc=edu
> ou: Users
> objectclass: top
> objectclass: organizationalUnit
> EOF
adding new entry "ou=Users,dc=upf,dc=edu"
```

Configuració Samba

Configurarem el sistema per a que actuï com a servidor SAMBA. L'ús que volem donar a aquest servidor és únicament el d'oferir als usuaris la possibilitat de connectar-se a una unitat de disc remota on poder realitzar un petit backup de les dades més importants contingudes als seus ordinadors personals del despatx, així com també accedir a les seves dades al clúster (tot i que això ho poden fer igualment amb un client SCP). Degut a la poca ambició de l'ús que li volem donar al SAMBA, realitzarem una configuració força senzilla de la següent manera:

Modificar el servidor LDAP

Ja que el servei SAMBA utilitzarà els mateixos usuaris que el sistema, i aquests es troben a la base de dades de servidor LDAP, caldrà modificar-ne el servei per tal que s'adeqüi a les necessitats dels usuaris SAMBA. Per a aconseguir això, editarem el fitxer de configuració de LDAP `/etc/openldap/slapd.conf` i farem les següents modificacions:

- Carregar l'esquema predefinit per a usuaris de tipus SAMBA. Aquest esquema configura LDAP per a incloure tots els atributs que fan falta per a crear un usuari samba:

```
include         /etc/openldap/schema/samba3.schema
```

- Configurar l'accés als passwords de samba només per a l'usuari administrador de LDAP:

```
## Yast2 samba hack ACL
## allow the "ldap admin dn" access, but deny
## everyone else
access to attrs=SambaLMPassword,SambaNTPassword
    by dn="cn=Administrator,dc=upf,dc=edu" write
    by * none
## Yast2 samba hack ACL done
```

- Indexar la base de dades pels paràmetres típics de Samba per tal de millorar la velocitat d'accés a la informació dels usuaris:

```
index sambaSID eq
index sambaPrimaryGroupSID eq
index sambaDomainName eq
```

- Reiniciar el servei LDAP per a que utilitzi la nova configuració.

Configurar el servei SAMBA

Degut a la senzillesa dels requisits del nostre servei SAMBA (accés només al home dels usuaris, només dels usuaris existents a LDAP, compatibilitat amb quotes, i cap opció extra de impressores, i perfils d'usuaris de windows), optarem per una configuració molt simple. Per a això, editarem el fitxer de configuració de SAMBA `/etc/samba/smb.conf` i el configurarem de la següent manera:

- Secció global:

La configuració d'aquesta secció afectarà a tots els elements del servei samba. Aquesta secció disposa de moltes opcions, però com volem un servei força simple, no ens caldrà configurar-les. Per altra banda, les opcions de seguretat que el servei aplica per defecte són les idònies per al nostre servidor, de manera que tampoc caldrà definir-les. Sabent tot això, només caldrà configurar uns pocs punts.

  - Definició del nom del nostre grup de treball. Realment no en farem ús, però el definirem.
  - Establir LDAP com el mètode d'autenticació dels usuaris i configurar-ne el seu correcte accés.
  - El servei Samba per defecte sap gestionar bé les quotes dels usuaris. El problema que presenta és que les gestiona "internament", de manera que, si bé l'usuari mai podrà excedir la quota que se li ha assignat, no té manera de veure quanta està ocupant, ja que si demana l'ocupació de la unitat de xarxa, aquesta li retornarà l'espai total del disc (els usuaris tenen quotes de ~10GB i el disc té unes dimensions de ~1TB). Degut a això, caldrà que creem i definim un script que, al preguntar quant espai lliure té el disc, respongui amb les dimensions de la quota enlloc del disc.
  - La opció "`deadtime`" ens permet gestionar el temps que les connexions inactives es mantindran obertes. El valor per defecte és no tallar-les mai, i de moment ens sembla adequat, però la tindrem en compte en un futur en cas de tenir problemes.
  - Com a aquest servei només s'hi podrà accedir des de la xarxa interna i VPNs de la universitat, bloquejarem totes les connexions alienes. Normalment faríem això editant els fitxers `/etc/hosts.allow/deny`, però el daemon smbd no és compatible amb `tcp_wrappers`, de manera que ignora aquests fitxers. Tot i això, el servei permet editar aquestes opcions dins del fitxer de configuració.

```
[global]
     workgroup = IBE
     passdb backend = \
          ldapsam:"ldap://bhsrv2.upf.edu
          ldap://bhsrv1.upf.edu"
     ldap admin dn = cn=Administrator,dc=upf,dc=edu
     ldap group suffix = ou=Groups
     ldap idmap suffix = ou=Idmap
     ldap machine suffix = ou=Machines
     ldap passwd sync = Yes
     ldap suffix = dc=upf,dc=edu
     ldap user suffix = ou=Users
     idmap backend = ldap:ldap://bhsrv2.upf.edu \
          ldap:ldap://bhsrv1.upf.edu

     get quota command = \
          /home/homes/rocks/scripts/user_quota
     strict allocate = yes

     hosts allow =  193.145.57.0/24 \ # servidors
                    172.22.0.0/22 \   # xarxa upf
                    10.60.80.0/23 \   # vpn upf
                    10.8.0.0/24       # vpn dept.
     hosts deny = ALL
```

- Secció homes:

La configuració d'aquesta secció gestionarà l'accés al directori home dels usuaris. Per tal d'aconseguir això, només caldrà definir la opció "path", la qual estableix el directori que es servirà, amb la ruta al home de l'usuari, però, com tots els usuaris de samba seran a la vegada usuaris de Linux, la variable `%H` ens donarà directament el path correcte de l'usuari. Hem de garantir també que els permisos dels directoris i fitxers es mantindran consistents amb el sistema. Samba per defecte crea tots els directoris i fitxers amb permisos 755 (totals per l'usuari, i lectura i execució per al grup i la resta), i sota l'usuari amb què s'ha connectat al servei i amb el seu grup per defecte. Per a evitar això, utilitzarem les opcions `inherit acls` i `inherit pemissions`, per a forçar a que escrigui sota les condicions definides per a aquell directori des de Linux (suid, guid, sticky bit, heretant les acl per defecte, ...). Finalment replicarem aquesta configuració per als directoris scratch i backup_desk, degut a diferències de comportament entre clients amb Linux, Windows i MacOS.

```
[homes]
     comment = Home Directories
     path = %H
     username = %u
     read only = No
     inherit acls = Yes
     inherit permissions = Yes

[backup_desk]
     comment = Backups
     path = /home/backup_desk/%u
     username = %u
     read only = No
     inherit acls = Yes
     inherit permissions = Yes

[scratch]
     comment = Scratch
     path = /home/scratch/%u
     username = %u
     read only = No
     inherit acls = Yes
     inherit permissions = Yes
```

Configuració del firewall

Tot i que ja hem definit dins del fitxer de configuració des de quines subxarxes serà accessible el servei, configurarem també aquestes restriccions dins del firewall de sistema, ja que d'altra manera, les peticions arriben realment al servei i s'ha d'encarregar de denegar-les, generant una càrrega innecessària.

Script de quotes

Durant la configuració del servei hem decidit que utilitzaríem un script per a gestionar les peticions d'espai de disc, ja que, tot i que el home dels usuaris està limitat per quotes, el mètode per defecte no funciona bé en el nostre cas, ja que retorna l'espai total del disc. Des del home dels usuaris es té accés realment a 3 particions: homes, scratch i backups_desk, cadascuna d'elles amb quotes diferents (o sense elles en el cas de scratch). Donat que només podrem informar d'una sola quota, decidim utilitzar la quota de la partició backup_desk, ja que és la que en farà el principal ús. Finalment el script quedarà de la següent manera:

```
#!/bin/bash
PATH=/usr/bin:/usr/sbin:/bin
IAM=`id -un`IAM=$1
if [ $2 -lt 3 ]; then
     QUOT=`quota -u $3  -l | awk '{ if ($1 ==
          "/dev/mapper/vgbackup_desk-lvbackup_desk")(z =
          1); else if ( z == 1 ) {z = 0; print $3}}'`

     if [ $QUOT -gt 0 ]; then
          RES=`quota -v $3 -l | awk '{ if ($1 ==
          "/dev/mapper/vgbackup_desk-lvbackup_desk")
          (z = 1); else if ( z == 1 ) {z = 0; print
               "2 "$1" "$2" "$3" "$4" "$5" "$6" "$7"
                1024"}  }'`
          STRIPPED=`echo $RES | sed 's/*/ /g'`
          echo $STRIPPED
     fi
fi
```

Engegar serveis

Finalment, hauríem de configurar el servei per a que s'iniciï automàticament a l'engegar el sistema, però com aquest servei dependrà del clúster d'alta disponibilitat, configurarem el seu inici automàtic des d'allà.

Configuració Mysql

Per tal de configurar el servei de base de dades per als usuaris del sistema, utilitzarem MySQL. Degut a que aquest servei formarà part del clúster d'alta disponibilitat, la base de dades haurà d'estar situada en els volums que poden ser muntats per les màquines del clúster d'alta disponibilitat. Sabent això, de col·locar tots els fitxers de la base de dades, així com els fitxers de configuració utilitzant la següent estructura de directoris:

- `/sp/fs/mysql_user`
  - config

    Arxius de configuració del servei.
  - mysql

    Arxius de dades, logs, directori temporal del servei i sockets.

- scripts

  Scripts de gestió del servei.

- `/sp/fs/mysql_user_binlog`

  Arrel del volum que contindrà els fitxers de log binari de la base de dades. Contindrà un únic directori "bin-log" dins del qual s'hi emmagatzemaran automàticament aquests fitxers.

Fitxers de configuració

Editem el nostre fitxer `my.cnf` per a configurar el servei de manera que s'adeqüi a les nostres necessitats:

- Configurem els paràmetres de funcionament per defecte del servidor mysqld. Els canvis principals que farem seran el port (passem del port per defecte 3306 al 3307, d'aquesta manera evitarem molts atacs per força bruta des d'internet), modificarem també les rutes de les dades, sockets, logs i logs binaris per a adaptar-los a la nostra estructura.

```
[mysqld]
port    = 3307
socket  = /sp/fs/mysql_user/mysql/mysql.sock
log     = /sp/fs/mysql_user/mysql/logs/mysql_user.log
log-bin = /sp/fs/mysql_user_binlog/bin-log/mysql-bin
tmpdir  = /sp/fs/mysql_user/mysql/tmp
```

- Pel que fa a la resta de paràmetres del servidor hi deixarem les opcions per defecte, ja que aquest pretén ser un servidor de propòsit general i, de moment, no tenim cap necessitat específica.

```
back_log = 50
max_connections = 100
max_connect_errors = 10
table_cache = 2048
max_allowed_packet = 512M
binlog_cache_size = 1M
max_heap_table_size = 64M
sort_buffer_size = 8M
join_buffer_size = 8M
thread_cache_size = 8
thread_concurrency = 8
query_cache_size = 64M
query_cache_limit = 2M
ft_min_word_len = 4
default_table_type = MYISAM
thread_stack = 192K
transaction_isolation = REPEATABLE-READ
tmp_table_size = 64M
long_query_time = 2
log_long_format
server-id = 7
relay-log=bhsrv-relay-bin
key_buffer_size = 32M
read_buffer_size = 2M
read_rnd_buffer_size = 16M
bulk_insert_buffer_size = 64M
myisam_sort_buffer_size = 128M
myisam_max_sort_file_size = 10G
myisam_max_extra_sort_file_size = 10G
myisam_repair_threads = 1
myisam_recover
[mysqldump]
quick
max_allowed_packet = 512M
[mysql]
no-auto-rehash
[isamchk]
key_buffer = 512M
sort_buffer_size = 512M
read_buffer = 8M
write_buffer = 8M
[myisamchk]
key_buffer = 512M
sort_buffer_size = 512M
read_buffer = 8M
write_buffer = 8M
[mysqlhotcopy]
interactive-timeout
[mysqld_safe]
open-files-limit = 8192
```

Degut a que el fitxer de configuració `/sp/fs/mysql_user/config/my.cnf` només serà visible des del node que tingui muntat el servei en aquell moment, caldrà editar el fitxer de configuració per defecte del sistema `/etc/my.conf` per tal de modificar el comportament del client mysql.

- Configurem els paràmetres de funcionament del client mysql: el socket i ports a utilitzar per defecte. Cal destacar que en el client configurarem com a port per defecte el port predeterminat de mysql 3306, i no el 3307 que hem escollit per al servidor, ja que la comunicació interna com a localhost es realitzarà pel socket enlloc de pel port. D'aquesta manera també es simplificaran les connexions a l'exterior en cas de ser necessàries.

```
[client]
port            = 3306
Socket          = /sp/fs/mysql_user/mysql/mysql.sock
```

Crear la base de dades

Amb els passos realitzats anteriorment ja hem configurat tant el client com el servidor mysql, però com ens trobem davant d'una nova instal·lació, caldrà inicialitzar la base de dades (concretament, la base de dades "mysql" que gestiona el funcionament del servidor). Per tal d'assolir això, utilitzarem un dels scripts que venen amb la distribució de mysql per tal de crear una base de dades inicial:

```
mysql_install_db --user=mysql
     --ldata=/sp/fs/mysql_user/mysql/data/
```

Un cop realitzat això, utilitzarem la comanda "`mysqladmin`" per a crear l'usuari inicial des amb el que poder connectar a la base de dades.

```
mysqladmin -u root password xxxxxxx
```

Engegar el servei.

Un cop fets tots aquests passos, ja tenim una base de dades plenament funcional, de manera que ja podem engegar el servei. Com caldrà que iniciem el servei utilitzant el sistema d'alta disponibilitat i utilitzant uns fitxers de configuració que no són els que usa per defecte, caldrà modificar el script que engega el servei, `/sp/fs/mysql_user/scripts/mysql.sh`, deixant-lo de la següent manera:

```
#! /bin/sh

mycfg=/sp/fs/mysql_user/config/my.cnf
databin=/sp/fs/mysql_user/mysql/data
soft=/usr/bin
pid_file=${databin}/`hostname`.pid


case "$1" in
   start)
      echo $echo_n "Starting MySQL"

      cd ${databin}
      ${soft}/mysqld_safe --defaults-extra-file=$mycfg
-u mysql --datadir=$databin
--log_error=${databin}/`hostname`.err
--pid-file=${pid_file} >/dev/null 2>&1 &
      ;;
   stop)
      echo $echo_n "Shutting down MySQL"
      kill `cat $pid_file`
      if [ -f $lock_dir ]; then
         rm -f $lock_dir
      fi
      ;;
  restart)
      $0 stop
      $0 start
      ;;
  *)
     echo "Error en els parametres"
     echo "Usage: $0 [start|stop|restart]"
     ;;
esac
```

Configuració del firewall

Inicialment, a aquest servei només s'hi haurà d'accedir des dels servidors que formin part del clúster, ja sigui des del clúster de càlcul, o per alguna aplicació web des del clúster d'alta disponibilitat. Degut a això, editarem tant el firewall com el fitxer `/etc/hosts.allow` i donarem accés només a aquestes màquines (bhsrv1 & 2, bhfront i tota la subxarxa 172.22.201.0/24).

Configuració Servei Web

Instal·larem al sistema un servidor web. El principal ús que li donarem serà el d'oferir les pàgines web acadèmiques dels usuaris del sistema. Aquestes webs seran gestionades directament pels usuaris i en general estaran realitzades en html pla o fins i tot seran índexs (utilitzant la opció d'apache +Indexes) que utilitzaran com a repositoris de dades, tot i que també els permetrem utilitzar diversos llenguatges de programació web o CMS. Tot i que aquest és l'objectiu inicial, no descartem que en un futur oferim també altres serveis, però com inicialment els desconeixem, configurarem el sistema únicament per a que serveixi els homes dels usuaris responent a la ip flotant. Com el sistema estarà en alta disponibilitat, utilitzarem la següent estructura de directoris compartits:

- `/homes/rocks/apache`
  - logs

    Logs d'accés i error dels serveis web que oferim.
  - vhosts

    Fitxers de definició i configuració de tots els hosts que servim.
- `/homes/rocks/www`

  Directori que servirà com arrel a tots aquells sites que servim i no pertanyin a usuaris.
- `/homes/users/*/public_html`

  Directori on trobarem la web personal de cada usuari.

Com l'ús que farem del servei inicialment serà molt limitat i no tenim cap necessitat

especial de configuració, l'únic canvi que farem en els fitxers de configuració serà crear un link que redirigeixi el directori `vhosts.d` (on es situen els fitxers de configuració de cadascun dels serveis diferenciats que gestionarem individualment) cap al seu equivalent a la partició homes.

Crearem els fitxers de configuració dels diferents hosts virtuals que necessitarem. Com inicialment només volem servir els homes dels usuaris, i només des de les peticions que ens arribin utilitzant la url de la ip flotant (bhusers.upf.edu enlloc de bhsrv1 o bhsrv2), necessitarem configurar els hosts virtuals de la següent manera:

- Creació d'un host virtual per defecte que denegui totes les peticions al fitxer `/homes/rocks/apache/vhosts.d/default.conf`. L'únic requisit d'aquest host virtual és que no faci res, denegant totes les peticions i guardant-ne un log a la partició compartida, quedant el fitxer de configuració d'aquesta manera:

```
DocumentRoot "/home/homes/rocks/www"

ErrorLog /home/homes/rocks/apache/logs/default-error_log
CustomLog
     /home/homes/rocks/apache/logs/default-access_log \
     combined

<Directory "/">
        Order Deny,Allow
        Deny from all
</Directory>
```

- Creació d'un host virtual que serveixi els homes dels usuaris només per a les peticions que hi arribin utilitzant com a url "bhusers.upf.edu". Les peticions que arribin utilitzant aquesta url però que no utilitzin la forma d'accés als homes dels usuaris ("`~username`" o "`user/username`") seran redirigides a la web del departament:

```
NameVirtualHost bhusers.upf.edu:80

<VirtualHost bhusers.upf.edu:80>

 ServerAdmin mail@upf.edu
 ServerName bhusers.upf.edu
 ServerAlias bhusers.upf.edu
 DocumentRoot "/home/homes/rocks/www/bhusers/"
 ErrorLog /homes/rocks/apache/logs/bhusers-error_log
 CustomLog /homes/rocks/apache/logs/bhusers-access_log \
                    combined

 <Directory "/homes/rocks/www/bhusers">
        Options +FollowSymLinks
        AllowOverride All
        Order Deny,Allow
        Allow from all
        Redirect Permanent / http://www.upf.edu/bioevo/
 </Directory>


 <IfModule mod_userdir.c>
        UserDir public_html
        Include /etc/apache2/mod_userdir.conf
        AliasMatch ^/users/([a-zA-Z0-9-_.]*)/?(.*) \
                    /homes/users/$1/public_html/$2

        <Directory "/homes/users/*/public_html">
                Options +FollowSymLinks
                AllowOverride All
                Order Deny,Allow
                Allow from all
                Options Indexes
         </Directory>

         <Directory "/">
                Options +FollowSymLinks
                Order Deny,Allow
                Allow from all
         </Directory>
 </IfModule>

</VirtualHost>
```

- Finalment obrirem el servei a l'exterior. Configurem el firewall per a permetre l'accés al protocol http per totes dues interfícies, i no cal que editem el fitxer `hosts.allow` ja que httpd2 no és compatible amb `tcp_wrappers`.

Configuració Alta disponibilitat

Un cop hem configurat tots els serveis que oferiran els nodes del clúster d'alta disponibilitat, només ens quedarà configurar el sistema per a que realment ofereixi aquesta alta disponibilitat. Existeixen diverses eines per a aconseguir això, però nosaltres utilitzarem *Heartbeat*, dins del projecte "*High-Availability Linux*" (o HAL).

HAL inclou un conjunt de daemons els quals s'encarreguen de comprovar contínuament l'estat de totes les màquines que formen part del clúster, així com dels serveis que aquestes ofereixen, controlant-ne el seu correcte funcionament i actuant (alertant i reiniciant o movent el servei de màquina) en cas de caiguda del servei o problema amb les comunicacions, emprant diverses tècniques com *Quòrum* (consens entre les màquines "supervivents") o *Fencing* (aïllar les màquines fora de control per tal d'evitar que prenguin el control de recursos del clúster si es considera que la màquina no està actuant bé). La configuració de tots aquests serveis és força senzilla i intuïtiva, ja que per a activar el servei només caldrà configurar la connectivitat entre tots els nodes, i després configurar cadascun dels serveis d'una forma molt similar als scripts de d'inici del sistema de init.d.

Configuració del servei LinuxHA

Per a configurar inicialment el funcionament del servei, haurem de definir-ne la connectivitat entre els nodes del clúster en diversos nivells:

- Hosts i interfícies:

Cal definir quins hosts formaran part del clúster d'alta disponibilitat, i per quina via es comunicaran. Això és possible fer-ho definint explícitament el nom dels hosts que formaran el clúster d'alta disponibilitat (opció recomanada si tenim un nombre controlat de màquines que no ha de variar), o bé utilitzant la opció "autojoin" per a permetre que noves màquines de la xarxa s'uneixin al clúster (recomanat si tenim un nombre de servidors variable). També caldrà definir les interfícies de comunicació entre els nodes. La documentació del servei recomana utilitzar, en sistemes en què els nodes tenen una interfície connectada a una xarxa "pública" a través de la qual ofereixen els seus serveis

i una altra a una xarxa "privada" d'ús exclusiu dels nodes del clúster, la comunicació broadcast per la xarxa "privada", i la multicast per la xarxa "pública". En el nostre cas, però, com els servidors ofereixen serveis per totes dues interfícies, utilitzarem preferentment la comunicació multicast a totes per totes dues xarxes, però hi configurarem també la broadcast per si es dónes el cas que les dues multicast fallessin.

```
udpport 694
autojoin none
mcast eth0 224.0.0.2 694 2 0
mcast eth1 224.0.0.2 694 2 0
bcast eth0
bcast eth1
node bhsrv2
node bhsrv1
```

- Protocol de comunicació:

Un cop hem definit per quina via es comunicaran els hosts, caldrà definir les característiques d'aquesta comunicació, especialment els llindars dels timeouts que utilitzarà el sistema per a decidir si hi ha problemes amb els hosts. Definirem la política de la següent manera:

  - Temps entre "senyals de vida": 1 segon.
  - Temps entre senyals per a "warning": 2,5 segons (2 senyals perdudes).
  - Temps per a donar un node per "mort": 4 segons (3~4 senyals perdudes).
  - Temps d'espera a l'arrencada: 12 segons (ajustat experimentalment).

Configurarem també l'ús del protocol "crm" per a replicar la informació de la configuració entre els nodes un cop el sistema està en marxa, i donarem el path del dispositiu de watchdog que permet al sistema d'alta disponibilitat disposar d'un recurs addicional en cas de necessitar reiniciar la màquina per mal comportament (linux-HA ja disposa dels seus propis mecanismes, configurar el watchdog ofereix redundància).

```
keepalive 1000ms
warntime 2500ms
deadtime 4000ms
initdead 12000ms
crm true
watchdog /dev/watchdog
```

Un cop configurats tots aquests paràmetres inicials, ja és possible iniciar el servei d'alta disponibilitat. Modificarem la configuració del firewall per tal d'obrir el port necessari per a la comunicació entre nodes (694 UDP) per totes dues interfícies. Finalment, configurem l'usuari d'accés "hacluster" i utilitzem el script "`ha_propagate`" que ve amb la distribució per a copiar aquesta configuració a tots els hosts. Ara ja podem engegar la GUI del "heartbeat" (el daemon encarregat de manegar els serveis) i acabar de configurar l'alta disponibilitat.

Configuració de la resta de paràmetres

Un cop accedim a la GUI, se'ns dóna opció a configurar una sèrie de paràmetres sobre el comportament del sistema en cas de caiguda. Aquest paràmetres seran:

- No Quorum Policy : stop. Quan cau alguna de les màquines, en cas de no haver-hi consens entre les màquines supervivents, parar els serveis de la màquina caiguda (mai tindrem quòrum ja que només tenim dues màquines, i quan una cau només ens queda l'altra per a prendre les decisions).
- Symmetric Cluster: yes. Tenim un clúster simètric amb màquines d'igual potència.
- Stonith Enabled: yes. Habilitar el "Stonith" (*Shoot The Other Node In The Head*) automàtic de les màquines que no actuïn bé.
- Stonith Action: reboot. En cas que es donin les condicions de "stonith", actuar reiniciant l'altre node.
- Default Resource Stickiness: 0 / Default Resource Failure Stickiness: 0. Tendència per defecte dels recursos a mantenir-se en el node en què estan. Normalment seria bo activar-ho, però definirem nosaltres mateixos la "stickiness" dels serveis, i això n'alteraria el funcionament.
- Stop Orphan Resources/Actions: yes. Aturar elements actius dins d'un servei que es dóna per desactivat.

- Startup Fencing: yes.
- Start Failure is Fatal: yes.

Configuració dels serveis:

Finalment, un cop ja hem configurat el funcionament del propi gestor d'alta disponibilitat, podrem definir els serveis que oferirà aquest clúster, els elements que el conformen, i com es comportaran. Els serveis que oferirem a través de l'alta disponibilitat es defineixen a través de "grups de recursos", els quals estan formats per una llista ordenada de recursos que seran engegats o apagats en ordre (o no) a l'activar el servei. Aquestes llistes de recursos, són molt similars al mètode típic de Linux d'engegar serveis a l'iniciar el sistema, consistint en crides ordenades a un script o acció (de fet, la majoria de recursos són els propis scripts que trobem a `/etc/init.d`, més uns pocs scripts propis de HAL), cridant-los amb el paràmetre "start" a l'engegar-los i "stop" a l'apagar-los (si el grup de recursos està ordenat, les crides a l'apagar el servei es fan en ordre invers al d'inici). Els serveis que hem d'oferir són:

- volum nfs homes
- volum nfs scratch
- volum nfs backup_desk
- volum mysql
- volum mysql_binlog
- apache
- samba
- mysql

Podríem oferir cadascun d'aquests serveis creant un grup de recursos independent per a cadascun d'ells, però no és possible fer-ho, ja que hem de tenir en compte que cadascun dels serveis s'iniciaria i apagaria de manera independent de la resta, i podria moure's d'un node a l'altre "lliurement" (podem posar-hi restriccions, però la idea és que tots dos nodes puguin muntar tots els serveis, tot i que no a la vegada). Això ens pot portar a problemes ja que tenim un seguit de dependències entre serveis:

- El volum homes conté links als volums scratch i backup_desk. Això implica

que el node que munti el volum homes, com a mínim ha de tenir accés a aquests volums.

- El servei apache ha de tenir accés als volums homes, scratch i backup_desk.
- El servei samba ha de tenir accés als volums homes, scratch i backup_desk.
- El servei mysql ha de tenir accés al volum mysql i mysql_binlog.

Per a resoldre el primer punt, l'accés des de homes als volums scratch i backup_desk, és necessària una complexa estructura de links la qual ens asseguri que és possible accedir al volum homes a través de `/home/homes` i `/homes`, al volum scratch a través de `/home/scratch` i `/scratch`, i al volum backup_desk a través de `/home/backup_desk` i `/backup_desk`, en tot moment i estiguin en el node que estiguin els serveis. Això seria possible creant per a cadascun dels serveis, un seguit de scripts que s'encarreguessin de, en el moment d'iniciar el servei i muntar el volum en un node, generar tots els links necessaris dins d'aquell node, i muntar la mateixa partició via NFS a l'altre node a la vegada que hi crea també l'estructura de links. Aquests scripts, a més a més, hauran de controlar la correcta neteja i "higiene" de l'estructura de directoris a tots dos nodes en el moment que el servei s'apaga en un node (per el motiu que sigui, sia migració o caiguda del node). Aquesta tasca, tot i que complicada, és possible realitzar-la, però presenta tres grans inconvenients:

- No ens garanteix el correcte estat de l'estructura de directoris en cas de caiguda inesperada d'un node, ja que no s'han pogut executar els scripts de neteja.

- El fet de que cada node hagi de muntar via NFS els volums que no estigui muntant ell directament, provoca que si, suposant que el node1 té muntada la partició homes i el node2 la partició scratch, un usuari a través de samba es connecta al seu home i des d'allà intenta accedir al directori scratch, això resoldrà en l'usuari fent una petició samba al node1, que a la seva vegada ha de fer una petició NFS al node2, el qual és qui té el disc muntat. I el mateix passaria amb l'accés web, i les peticions de disc del clúster d'alta disponibilitat. D'aquesta manera, si els volums estan muntats en nodes diferents, totes les peticions provocaran càrrega a tots dos nodes enlloc de a un sol.

- Fa molt complicada l'ampliació futura. Si en un futur volem afegir un nou volum al sistema o afegir-hi un nou node d'alta disponibilitat, hauríem de refer tots els scripts tenint en compte els canvis.

Veient tot això, decidirem seguir una via més conservadora i, en lloc de disposar de 7 serveis, en tindrem només dos: el servei d'usuaris (volums homes, scratch i backup_desk, i serveis samba i web) i el servei mysql (volum mysql i mysql_binlog i servei mysql). D'aquesta manera, no tindrem els problemes amb els links que ens provoca el mètode anterior, ja que els links els crearem a l'arrel del disc local de tots els nodes, de manera que el node que munti els volums els tindrà disponibles, i l'altre node tindrà uns links trencats però que no farà servir per a res (fins que el servei no migri cap allà pel motiu que sigui). A més a més, tot i que tindrem la gran majoria de la càrrega del disc en un sol node, ens assegurem el funcionament del sistema, ja que l'objectiu real d'aquest clúster d'alta disponibilitat no és la d'actuar com a balancejador de càrrega, sinó la d'oferir alta disponibilitat dels discs per al clúster de càlcul.

Aquests dos "grans serveis" els definirem en dos grups de recursos de la següent manera:

- User_server: servei que oferirà tot els serveis que utilitzin els volums homes, scratch i backup_desk.

Servei	Descripció
user_server_begin	Mail d'advertència per l'inici/final del moviment servei (depenent de si està engegant o parant serà el primer o últim en executar-se).
SuSEfirewall_restart_init	Reinici del firewall, necessari per al correcte funcionament de les quotes amb NFS, ja que el port del rquotad va a través del portmapper, però el firewall només obre els ports que el portmapper té actius en el moment en què s'engega el firewall. Cal posar aquest servei als dos extrems del grup per tal que s'executi al final tant de l'engegada com de la parada del servei i així assegurar que els ports només estan oberts quan és necessari.

scratch_fs	Muntatge de la partició scratch.
homes_fs	Muntatge de la partició homes.
backup_desk_fs	Muntatge de la partició backup_desk.
portmap	Daemon per a gestionar els ports dels serveis necessaris per al servidor NFS.
nfsserver	Servei NFS.
idmapd	Resolució de noms per NFSv4.
quotes	Activar quotes en aquest servidor i quotes remotes per a NFS.
samba	Servei samba.
homes_ip_pub	Captura de la ip flotant de la xarxa pública que respon a bhusers.upf.edu (193.145.57.223).
homes_ip_priv	Captura de la ip flotant de la xarxa privada que respon a bhusers.s.upf.edu (172.22.201.223).
scratch_ip_pub	IP pública bhscratch.
scratch_ip_priv	IP privada bhscratch.
web_users	Servei apache.
SuSEfirewall_restart_end	Reinici del Firewall.
user_server_end	Mail d'avís del fi de l'engegada/inici de la parada del servei.

- Mysql_user: grup que oferirà tot el necessari per a l'ús del servei mysql.

Servei	Descripció
mysql_user_begin	Mail de confirmació de l'inici de l'engegada/fi de la parada del servei.
mysql_firewall_restart_init	Reinici del firewall. Si es crida amb "start" obre també el port de mysql, si es crida amb "stop", no.
fs_mysql_user	Muntatge del volum mysql.
fs_mysql_user_binlog	Muntatge del volum mysql_binlog.
mysql_user_sw	Inici servei mysql.

mysql_user_ip_priv	Captura de la ip flotant de la xarxa privada que respon a bhmysql-user.s.upf.edu (172.22.201.225)
mysql_user_ip_pub	Captura de la ip flotant de la xarxa pública que respon a bhmysql-user.upf.edu (193.145.57.225).
mysql_firewall_restart_end	Reinici del firewall. Si es crida amb "start" obre també el port de mysql, si es crida amb "stop", no.
mysql_user_end	Mail d'avís del fi de l'engegada/inici de la parada del servei.

Un cop configurat això, ja tenim tots els serveis d'alta disponibilitat muntats i funcionant. Afegirem al sistema però, unes "*constraints*" que facin que cadascun dels dos serveis tinguin tendència a funcionar en un dels dos nodes, de manera que distribuïm una mica la càrrega, però permetent el moviment de serveis d'un node a l'altre en cas de necessitat.

Seguretat del sistema

Degut a que els nodes que formen el clúster d'alta disponibilitat estaran connectats directament a internet i disposaran d'una IP pública per a accedir-hi, caldrà protegir-los per a evitar problemes, com accessos no desitjats o explotació de vulnerabilitats de serveis que ni tan sols utilitzem. Sabent això, la política que utilitzarem serà tancar tots els ports i obrir només aquells necessaris pels serveis que utilitzarem.

Firewall

- Xarxa externa:
  - No enrutarem els paquets des de la xarxa externa cap a la interna.
  - Ports TCP oberts: ldap, ldaps, ssh. Obrirem el port http i mysql només si el servei està actiu en aquesta màquina (s'encarregaran els scripts que crida el daemon d'alta disponibilitat).
  - Ports UDP oberts: cap.
  - Serveis RPC: mountd, nfs, nfs_acl, nlockmgr, portmap, status.

- Xarxa interna:
    - TCP: ldap, ldaps, ssh, samba (microsoft-ds, netbios-ssn). Obrirem el port http i myqsl només si el servei està actiu en aquesta màquina.
    - UDP: samba (netbios-dgm netbios-ns), ntp i 694 (heartbeat).
    - RPC: mountd, nfs, nfs_acl, nlockmgr, portmap, status.
- Xarxes de confiança:
    - Habilitarem l'accés als ports de samba (tcp microsoft-ds i netbios-ssn, i udp netbios-dgm i netbios-ns), a les subxarxes següents:
        - Nodes del clúster d'alta disponibilitat.
        - Front-end del clúster de càlcul.
        - Xarxa PCs dels despatxos.
        - VPN.
    - Habilitarem l'accés al port 694 (comunicació alta disponibilitat) a la subxarxa dels nodes del clúster d'alta disponibilitat.

En resum, obrirem només els serveis necessaris dins la xarxa interna, i tindrem oberts per a tothom a l'externa els serveis ssh, http, mysql (als que s'ha de poder accedir des d'arreu), i ldap i nfs (als quals haurem de limitar-ne l'accés).

Configuració TCP WRAPPERS

Configurem la capa de seguretat tcp wrappers, la qual gestiona les peticions que pot acceptar un determinat servei depenent de l'adreça de la que provinguin. A través d'aquesta capa, limitarem l'accés als serveis ldap i nfs, els quals permetrien l'accés a qualsevol usuari utilitzant només la configuració del firewall.

Inicialment denegarem l'accés a tots els serveis ficant "ALL: ALL" al fitxer `/etc/hosts.deny`, i configurarem el fitxer `/etc/hosts.allow` de la següent manera:

- Nodes del clúster d'alta disponibilitat i clúster de càlcul:
    - NFS (portmap, lockd, mountd, rquotad, statd).
    - LDAP (slapd).

- Tots els nodes:
    - Mysql.
    - sshd.

De cara tant a augmentar la seguretat del sistema com per a evitar que els usuaris puguin connectar-se als nodes del clúster d'alta disponibilitat, editarem el fitxer de configuració `/etc/security/access.conf` i hi afegirem les següents entrades per a prohibir l'accés de tots els usuaris, però permetre l'accés de determinats usuaris puntuals:

```
- : root : ALL
+ : theredia acarreno : ALL
- : ibe : ALL
```

Configuració fail2ban

Un cop configurat això, ja hem bloquejat l'accés a tots els serveis que no utilitzarem, així com l'accés des de "l'exterior" a serveis privats (tot i que funcionin per la xarxa externa). Els únics serveis que quedaran oberts a l'exterior seran SSH i mysql i apache només en el node que estigui oferint aquell servei.

Instal·larem als nodes del clúster d'alta disponibilitat, igual que instal·lem a tots els servidors del departament connectats a internet, el servei "fail2ban". Fail2ban és un daemon que es dedica a monitoritzar el contingut dels logs del sistema, i se'l pot configurar per a executar una determinada acció quan s'acompleix alguna condició. Degut a que freqüentment els nostres servidors es veuen sotmesos atacs en forma d'onades de peticions d'accés ssh des de l'exterior, l'ús que farem d'aquest servei (el mateix que en fem a la resta de servidors), serà llegir el log del sistema i, un cop detecti un determinat nombre d'intents de connexió erronis des d'una mateixa IP, bloquegi tots els paquets que arribin al servidor des d'aquesta IP. Per a configurar això, instal·larem el servei fail2ban editarem el fitxer `/etc/fail2ban/jail.conf` afegint-hi la següent entrada:

```
[ssh]

enabled  = true
filter   = sshd
action   = iptables[name=SSH, port=ssh, protocol=tcp]
logpath  = /var/log/messages
maxretry = 5
bantime  = 600
findtime  = 600
```

Aquesta configuració indica que filtrarem el fitxer `/var/log/messages` cercant totes les entrades marcades amb "sshd", i, si durant els últims 10 minuts (findtime) trobi 5 intents d'accés fallits des d'una mateixa IP, la bloquejarà (action) durant 10 mints (bantime).

Annex IV, instal·lació i configuració del clúster de càlcul

El clúster de càlcul estarà format per 9 hosts, 8 d'ells encarregats del càlcul i d'idèntica configuració, i un d'ells actuant com a punt d'accés i distribuïdor del càlcul entre la resta de nodes. D'ara en endavant utilitzarem la nomenclatura de Rocks-cluster i anomenarem "frontend" al node distribuïdor, i "compute" als nodes de càlcul.

Instal·lació del Sistema Operatiu

Al contrari del clúster d'alta disponibilitat, on hem instal·lat un sistema operatiu "típic" per a servidors com és SuSE Linux, per al clúster de càlcul utilitzarem un sistema operatiu especialment pensat per a aquest tipus de sistemes, Rocks-cluster, basat en CentOS (basat a la seva vegada en Red Hat Linux).

El principal avantatge que ens ofereix rocks respecte a les distribucions tradicionals és, a part de la obvia presencia i configuració dels paquets necessaris per al funcionament i gestió d'un clúster de càlcul, l'ús del sistema "Kickstart", el qual fa necessària només la instal·lació manual d'un sol node, mentre permet configurar i instal·lar automàticament la resta de nodes a través de la xarxa. Sabent això, per a instal·lar tot el clúster només caldrà instal·lar i configurar el frontend, i preparar des d'ell les imatges que carregaran automàticament els computes.

Instal·lació del frontend

Per a instal·lar el sistema operatiu del frontend, instal·larem rocks des dels discs de la distribució. La distribució rocks ve segmentada en diversos paquets independents, anomenats "*rolls*", els quals permeten la instal·lació de programes i serveis addicionals. La instal·lació bàsica de rocks requereix dels següents rolls:

- Kernel:
  Kernel de Linux

- OS (discs 1 i 2):
  Imatges dels discs d'instal·lació de CentOS 5 en què es basa rocks. Només son imprescindibles els discs 1 i 2, dels 7 que hi ha. Segons la documentació, és possible utilitzar els discs de Red Hat Linux 5 com a substitut.

- Base:
  Configuració i serveis bàsics per al clúster.

- Web-server:
  Servidor web necessari per a la comunicació entre el frontend i els computes.

Addicionalment als rolls mínims imprescindibles, és possible afegir a la instal·lació rolls addicionals que ofereix rocks. D'entre els diversos rolls que hi ha, decidim afegir a la instal·lació els següents:

- Area51:
  Paquet de seguretat que inclou diversos serveis, entre ells un analitzador de rootkits i tripwire, per a controlar les modificacions de fitxers de sistema.

- Ganglia:
  Sistema de monitorització i gestió de tots els nodes a través d'una interfície web.

- HPC:
  High-Performance Computing. Inclou sistemes de paral·lelització com OpenMP,

MPI,...

- Java:
  Instal·lació del sdk de java.

- SGE:
  Instal·lació del software de gestió de recursos Sun Grid Engine.

Rocks també ofereix altres rolls interessants que ens hem plantejat instal·lar però que finalment hem descartat:

- Bio:
  Conté diversos programes utilitzats en bioinformàtica, però són molt pocs i en necessitarem instal·lar molts més, de manera que optarem per no utilitzar aquest roll i gestionar nosaltres mateixos la instal·lació de tot el software necessari.

- Condor:
  Sistema de gestió de recursos per a clústers de tipus grid molt heterogenis. Sembla estar més pensat per a treballar amb workstations que amb clústers altament integrats com el nostre, de manera que, de moment, no l'utilitzarem.

- Torque:
  Sistema de gestió de recursos juntament amb el programador de jobs Maui. Fa la mateixa funció que SGE, de manera que només caldrà instal·lar un dels dos.

- Pvfs2:
  Sistema de fitxers distribuït. És útil en altres tipus d'instal·lacions, però en la nostre, amb un servidor de disc centralitzat i nodes amb discs de poca capacitat, no resulta útil.

Un cop seleccionats els rolls que instal·larem, procedirem a configurar la resta de la instal·lació:

- Interfícies de xarxa:

La instal·lació de rocks-cluster requereix que tots els nodes, tant computes com frontend, estiguin connectats a través d'una xarxa ethernet d'ús exclusiu per a rocks, així com a mínim d'una altra interfície de xarxa per al frontend, a través de la qual tingui accés a l'exterior. Sabent això, configurarem les interfícies de la següent manera:

- eth0:
  - Connectada a la xarxa 172.22.205.0/24, interna i d'ús exclusiu pels nodes del clúster.
- eth1:
  - Connectada a la xarxa 172.22.202.0/24, interna però accessible des de la resta de xarxes de la universitat (PC's de sobretaula inclosos) i amb accés a internet a través del proxy.

- Data i hora:

  Configurem el servidor NTP per a sincronitzar l'hora del sistema.

- Particionat del disc:

  Els requisits mínims per a instal·lar rocks tant pel frontend com pels computes és de 20 GB. El particionat per defecte que ofereix rocks és el següent:

Partició	Espai	Sistema de fitxers
/	8 GB	Ext3
/var	4 GB	Ext3
Swap	1 GB	Swap
/state/partition1	La resta (54 GB)	Ext3

Aquest particionat ens sembla bàsicament adequat, però hi farem unes lleugeres modificacions:

- L'espai de `/var` al frontend hauria de ser més gran, ja que els seus logs recolliran també els de la resta de nodes
- Hem d'ampliar la swap a tots els nodes, ja que tots els blades tindran com a mínim 9GB de memòria, amb lo qual, 1GB de swap ens resulta pràcticament inútil (en l'instant que el node comenci a fer swapping, de seguida l'omplirà), de

manera que ampliarem la swap del frontend a 9GB (hauria de ser suficient ja que aquest node mai hauria de fer un ús intensiu de la memòria).

Amb això, la taula de particionat quedarà de la següent manera:

Partició	Espai	Sistema de fitxers
/	8 GB	Ext3
/var	8 GB	Ext3
Swap	9 GB	Swap
/state/partition1	La resta (42 GB)	Ext3

Un cop establerts tots els paràmetres de la instal·lació, aquesta s'efectuarà automàticament. Un cop finalitzada, passarem a configurar el frontend, però abans, realitzarem una instal·lació bàsica per a tots els computes per tal de integrar-los al sistema i així poder comprovar el funcionament del serveis.

Instal·lació inicial bàsica dels computes

Per tal de poder comprovar el correcte funcionament dels serveis mentre els configurem al frontend, caldrà que el sistema hagi detectat i configurat els nodes de càlcul. La instal·lació d'aquests nodes es farà automàticament i de manera transparent per a nosaltres, de manera que l'únic pas en què haurem d'intervenir és en la detecció inicial dels nodes.

Per tal de realitzar aquesta detecció i instal·lació de nous nodes, utilitzarem la utilitat inclosa amb la distribució "`insert-ethers`". Aquesta utilitat el que fa és detectar les peticions DHCP que realitzen els nodes a l'engegar en mode PXE (arrancar per xarxa), desa la MAC del node i li assigna automàticament un hostname (amb la forma compute-X-Y on X és el número de la cabina i Y el número de node) i una IP de la xarxa privada (172.22.205.0/24) les quals desa a la base de dades de rocks per a mantenir-lo encara que es reinicïi el compute. Per a detectar tots els nodes que actuaran com a compute, només caldrà que executem com a root "`insert-ethers`", seleccionem "Compute"

com al tipus d'instància al que pertanyeran els nodes que detecti, i engeguem per ordre un a un tots els nodes fins que el programa hagi detectat la MAC de l'últim node.

Aquesta utilitat però, té l'inconvenient que només configura els nodes amb la interfície eth0 connectada a la xarxa de comunicació de rocks, però en el nostre cas necessitarem també configurar la interfície eth1 per a poder accedir als serveis del clúster d'alta disponibilitat. Per a solucionar això, un cop instal·lats els computes, només caldrà que afegim la informació sobre la interfície eth1 de cada compute a la base de dades de rocks amb la següent comanda:

```
rocks set host interface ip NODENAME eth1 IP
rocks set host interface name NODENAME eth1 \
     NODENAME.s.upf.edu
rocks set host interface subnet NODENAME eth1 public
```

i reinstal·lar-la de nou fent:

```
ssh NODENAME "/boot/kickstart/cluster-kickstart"
```

Un cop fet això, ja disposem d'un clúster funcional. Ara només quedarà adaptar-ne la configuració a les nostres necessitats específiques.

Configuració del clúster

Ara ja tenim instal·lats tant el sistema operatiu com els diferents serveis integradors del clúster, però ens caldrà adaptar aquesta configuració. Això ho durem a terme en dues fases: primer configurarem el sistema al node que actuarà com a front-end, i un cop aquest estigui completament preparat, configurarem la imatge que serveix als nodes de càlcul i els reiniciarem per a que s'instal·lin ja configurats automàticament.

Configuració dels elements bàsics del S.O.

Hosts

El primer que caldrà configurar al sistema serà una petita "trampa" al fitxer de hosts per a fer que el sistema funcioni correctament: Tots els nodes del clúster de càlcul han d'accedir als serveis hostatjats a ambdós nodes del clúster d'alta disponibilitat. Degut a les lleugeres diferències que hi ha entre els nodes de càlcul i el front-end, concretament en la connexió a xarxes diferents a través de la interfície eth1, serà necessari que els nodes de càlcul i el front-end es connectin de manera diferent als serveis d'alta disponibilitat. Concretament, els nodes de càlcul accediran directament a través de la xarxa 172.22.201.0/24, mentre que el front-end hi accedirà a través de la 172.22.202.0/24, enrutant posteriorment a la xarxa externa 193.145.57.0/25.

Rocks utilitza el daemon 411 per a propagar i mantenir sincronitzats a tots els nodes un seguit de fitxers de configuració, entre ells els encarregats de la configuració dels volums de disc NFS. Com aquests fitxers hauran de ser idèntics per a tots els nodes, tot i que el front-end hagi d'utilitzar una IP diferent que els nodes de càlcul, caldrà trobar una solució.

Configurarem tots els serveis que hagin d'utilitzar IPs diferents per a nodes i front-end (és a dir, IPs de la xarxa 172.22.201.0/24), de manera que utilitzin explícitament la configuració correcta per als nodes de càlcul. Això farà que els serveis no funcionin al front-end, cosa que solucionarem afegint un àlies al fitxer de hosts del node, de tal manera que totes les peticions que hagi d'enviar a través de la xarxa 172.22.201.0/24 (*.s.upf.edu) siguin redirigides a la xarxa 193.145.57.0/25 (*.upf.edu). Hem de tenir en compte, però, que algunes de les comandes que inclou la distribució per a gestionar la inserció i eliminació de nodes del clúster modifiquen el fitxer de hosts, de manera que afegirem un script al crontab que s'encarregarà de restaurar el fitxer de hosts amb el "hack" quan detecti que s'ha produït una modificació del fitxer.

Usuaris

Rocks-cluster ve configurat inicialment per a mantenir actualitzats els usuaris i grups de totes les màquines mitjançant la còpia dels fitxers `/etc/passwd`, `groups` i `shadow` a tots els nodes del clúster, utilitzant el servei 411.d (el funcionament del qual detallarem més endavant). D'entrada no modificarem l'ús que es fa de la distribució d'aquests fitxers a través dels nodes, ja que comparteix de sèrie usuaris imprescindibles pel sistema operatiu (root, nagios, sge, ...), però pel que fa als usuaris "reals" del sistema, els obtindrem del servidor ldap que hem configurat al clúster d'alta disponibilitat.

La instal·lació per defecte de Rocks-cluster, inclou suport per a ldap i openldap, de manera que per a configurar l'accés al servidor només caldrà modificar-ne els fitxers de configuració. Per a fer-ho, enlloc de modificar directament els fitxers, utilitzarem la següent comanda:

```
/usr/sbin/authconfig --enableldap --enableldapauth
     --ldapserver=bhsrv2.s.upf.edu,bhsrv1.s.upf.edu
     --ldapbasedn="dc=upf,dc=edu" --update
```

Això ens deixarà configurat l'accés dels usuaris al front-end. Per a poder utilitzar els usuaris ldap a la resta de nodes, només caldrà executar automàticament aquesta mateixa comanda durant la fase de post-instal·lació del node.

Disc

La configuració del disc dels nodes es realitzarà en dues parts. Per una banda està el fitxer `/etc/fstab`, dins del qual rocks hi configura automàticament els volums resultants del particionat i formateig del disc durant la instal·lació del sistema, el qual no modificarem. Per altra banda, configurarem l'ús de les particions NFS a través del daemon autofs. Aquest daemon s'encarrega de muntar i desmuntar dinàmicament particions de disc segons si estan sent utilitzades o no. Aquest funcionament és ideal per a muntar sistemes de fitxers que depenguin d'un servidor d'alta disponibilitat, ja que, en cas de caiguda del servei o canvi de node, el disc es desmuntarà automàticament i tornarà a muntar, amb el servei inicialitzat a l'altre node.

La configuració de l'autofs s'estableix mitjançant el fitxer `/etc/auto.master`, el qual conté un llistat dels directoris on hi muntarem algun volum, així com els fitxers que contindran la configuració específica de cada directori. En el nostre cas, tindrà el següent aspecte:

```
$ cat /etc/auto.master

/home   /etc/auto.home   --timeout=1200
```

Aquesta configuració significa que només muntarem particions amb l'automounter dins del directori /home, definits dins del fitxer `/etc/auto.home` i que tindran un timeout de 20 minuts (el sistema de fitxers es desmuntarà automàticament passats 20 minuts des de l'últim accés). La configuració inicial de Rocks-cluster inclou també l'ús dels fitxers de configuració `auto.net`, `auto.smb` i `auto.web`, però com tots aquests serveis ens els donarà el clúster d'alta disponibilitat, els deshabilitarem. Pel que fa a la configuració de les particions del directori /home, les deixarem de la següent manera:

```
$ cat /etc/auto.home

homes          bhusers.s.upf.edu:/sp/fs/homes/&
backup_desk    bhusers.s.upf.edu:/sp/fs/backup_desk/&
scratch        bhusers.s.upf.edu:/sp/fs/scratch/&
install        bhfront.local:/export/home/&
```

Com veiem, el sistema muntarà automàticament dins del directori `/home` les particions homes, backup_desk i scratch, totes elles procedents del clúster d'alta disponibilitat. Notem que estem utilitzant la URI de la interfície interna, tal i com hem descrit anteriorment a l'apartat "hosts". Finalment també muntarem el directori `install`, el qual, tot i provenir directament del propi front-end, és necessari que muntem a tots els nodes ja que serà utilitzat pels nodes de càlcul durant la instal·lació inicial (o reinstal·lació deguda a caiguda o reinici inesperats) del seu sistema operatiu.

Tots aquests fitxers seran sincronitzats a tots els nodes del clúster de càlcul a través del daemon 411.d, per tal d'assegurar la idèntica configuració de tots els nodes.

Configuració dels serveis

NFS local

La distribució rocks-cluster configura automàticament un servei NFS al node que actuarà com a front-end. Per defecte, aquest servei es configura de tal manera que ofereix dos directoris del disc local, "`install`" i "`apps`", a tots els hosts connectats a qualsevol de les interfícies del front-end. La partició "`install`" té com a objectiu contenir i servir tots els elements necessaris per a la instal·lació i configuració del sistema operatiu dels nodes de càlcul. La partició "`apps`" té com a objectiu actuar com a punt de instal·lació, de manera que serveixi als nodes els diferents programes i scripts que els usuaris necessitaran executar. Aquesta configuració ens planteja dos problemes:

- El disc local dels nostres nodes és molt petit, i no ens servirà per a contenir-hi tots els programes que necessitaran els usuaris. Per a tal fita utilitzarem el volum "homes" que se'ns servirà des del clúster d'alta disponibilitat. Degut a això, desactivarem el volum "`apps`".
- Per defecte es servirà el volum per les subxarxes connectades a les dues interfícies ethernet del front-end. En el nostre cas només ens interessarà servir-ho a si mateix (pel motiu descrit a l'apartat disc) i per la subxarxa local del clúster, connectada a la interfície eth0, ja que la interfície eth1 dóna accés a la resta de la xarxa de la universitat, cap node de la qual necessita tenir accés a aquest volum.

Finalment, amb tot això, el fitxer de configuració dels volums servei NFS quedarà de la següent manera:

```
$ cat /etc/exports
/export/home 172.22.205.0/255.255.255.0(rw,async)
```

La resta de la configuració dels daemons del servei la deixarem igual.

Web

Rocks-cluster inclou en la instal·lació per defecte un servidor web apache. Aquest és necessari per al correcte funcionament de diversos serveis del sistema, com per exemple la transmissió de fitxers de configuració per al servei 411.d o la transmissió de les imatges de sistema per a la instal·lació automàtica de nodes, així com també sistemes d'administració dels nodes com és la interfície web del monitoritzador del sistema Ganglia.

Postfix

Per una banda, caldrà configurar el front-end per tal de permetre que enviï correus cap a l'exterior. Per això, caldrà modificar la configuració del postfix al fitxer /etc/postfix/main.cf i hi configurarem el relayhost el qual haurà d'apuntar al servidor de correu sortint de la universitat.

```
relayhost = mx.upf.edu
```

Per altra banda, degut a les necessitats del sistema de distribució de treballs que utilitzarem, quan un job finalitza al node de càlcul, aquest envia un correu amb destinatari "usuari@node.local". Per tal que aquest mail arribi a l'usuari, caldrà fer dues coses:

- Configurar el servei de correu del front-end per tal que reconegui com a propis els mails que emeten els nodes de càlcul. Per això, a part de reconèixer els mails que arribin dirigits al hostname, haurà d'acceptar també els mails dirigits a la resta dels àlies amb què treballarà el front-end, així com també els mails dirigits cap a qualsevol dels nodes de càlcul per les seves dues interfícies:

```
mydestination = $myhostname, cadaques.local, \
  bhfront.b.upf.edu, bhfront.local, localhost, \
  local, regexp:/etc/postfix/mydestination
```

Per tal de fer que el Postfix accepti els correus dirigits als nodes, amb adreces de la forma `*.local` i `*.s.upf.edu`, caldrà tractar-les com a expressions regulars a un fitxer de configuració a part, de la següent manera:

```
# cat /etc/postfix/mydestination

/.\.local$/ local
/.\.s\.upf\.edu$/ local
```

- Configurar el servei de correu per tal que relacioni els usuaris amb la seva adreça de correu "real", la informació sobre la qual es troba a la base de dades ldap. Per tal d'aconseguir això, caldrà fer-ho en dues fases: Primer caldrà crear un script el qual, utilitzant el llenguatge de comunicació amb LDAP, es connecti al servidor LDAP i faci una petició preguntant per l'usuari el nom del qual es correspon amb el de l'adreça de correu, i li respongui només l'adreça de correu "real":

```
server_host = bhsrv2.s.upf.edu bhsrv1.s.upf.edu

search_base = dc=upf, dc=edu
query_filter = uid=%s
result_attribute = mail
ldap_bind = no
```

Un cop tenim aquest script, caldrà editar la configuració del servei postfix per a afegir aquest script dins dels sistemes de mapping que farà servir per a identificar als usuaris del correu. Per a això, editarem un altre cop el fitxer `/etc/postfix/main.cf` i afegirem aquest script dins la opció "`alias_maps`", definint-la clarament com de tipus LDAP:

```
alias_maps = hash:/etc/aliases, \
                    ldap:/etc/postfix/ldap-aliases.cf
```

Un cop fet això, el servei de correu ja funcionarà perfectament redirigint tots els correus que arribin als usuaris del clúster cap a les seves adreces reals. Ara només caldrà modificar el fitxer `/etc/aliases` per a fer que tots els correus que siguin enviats cap a l'usuari root del front-end, siguin redirigits a l'adreça de correu d'administració que utilitzem al departament.

```
root:     mail@upf.edu
```

Variables d'entorn

Caldrà configurar algunes variables d'entorn del sistema, per tal de garantir el funcionament dels nodes de càlcul. Caldrà configurar especialment el proxy de sistema per a permetre que els nodes de càlcul accedeixin a internet (és necessari per a que alguns dels programes i scripts que executaran els usuaris puguin descarregar-se dades genòmiques des de servidors externs).

Degut a això, crearem el fitxer `/etc/profile.d/local.sh` dins del qual hi declararem totes aquelles variables d'entorn i àlies que calguin, i el sincronitzarem a tots als nodes mitjançant el daemon 411.d. Finalment, el fitxer quedarà de la següent manera:

```
# cat /etc/profile.d/local.sh

PS1="[\u@\h \W]\\$ "
PS1="[\u@\h \W]\\$ "
export http_proxy=http://proxy.upf.edu:8080
export https_proxy=http://proxy.upf.edu:8080
export ftp_proxy=http://proxy.upf.edu:8080
export no_proxy=localhost,172.0.0.0/8,193.145.57.0/24

export PATH=/aplic/perl5/bin:$PATH
export PATH=/aplic/python/bin:$PATH

export C_INCLUDE_PATH=/opt/openmpi/include: \
     /opt/mpich/gnu/include:/opt/mpich2/gnu/include
export CPLUS_INCLUDE_PATH=/opt/openmpi/include: \
     /opt/mpich/gnu/include:/opt/mpich2/gnu/include

alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
alias la='ls -la'
alias vi='vim'

alias RPMlist='rpm -qa | grep '
alias RPMinfo='rpm -ql '
alias bp='ps -ef | grep '
```

Totes aquestes variables d'entorn normalment les definiríem dins del fitxer `/etc/environment`, però en aquest cas serà millor tenir-les dins d'aquest fitxer, ja que durant la configuració del sistema gestor de cues SGE veurem que caldrà modificar el script de init que executa el daemon encarregat de la creació i execució dels jobs als nodes de càlcul, ja que, per defecte, durant la seqüència d'inicialització del sistema RC, les variables d'entorn no es carregaran, de manera que el daemon d'execució no utilitzarà aquestes variables, i els processos fills que executarà no les heretaran, de manera que, finalment, l'entorn en què s'executaran els jobs dels usuaris serà diferent al que es troba l'usuari al fer login al sistema. Per a solucionar això, modificarem al graph de kickstart el script que executa el daemon per a que inclogui una crida a `/etc/profile`, el qual carregarà automàticament tot l'entorn.

411.d

Rocks-cluster inclou per defecte el servei 411.d, el qual s'encarrega de la replicació automàtica de fitxers des d'un node cap a la resta. Aquest sistema es composa d'un script al qual se li passa un llistat de fitxers a compartir, els encripta i fa accessibles a tots els nodes de la subxarxa local a través del protocol http, i envia una senyal a través d'un canal multicast, advertint a tots els nodes que l'escoltin que s'ha produït una modificació en algun dels fitxers que serveix per a que els actualitzin. Aquest script es troba també al crontab, de manera que cada hora es comprovarà si hi ha hagut algun canvi.

Inicialment, el servei està configurat per a que repliqui únicament els fitxers d'usuaris (`/etc/passwd`, `groups` i `shadow`) i de disc (`/etc/auto.*`), però hi podem afegir tots aquells fitxers que modifiquem respecte a la configuració inicial del sistema i que ens interessi mantenir actualitzats als nodes de càlcul. Inicialment, per al correcte funcionament del sistema, afegirem a aquest servei la replicació dels fitxers `/etc/aliases` (per tal de resoldre els mails a root, tal i com hem descrit a l'apartat anterior), i el fitxer `/etc/logrotate.conf`, el qual s'encarrega de la rotació dels logs de sistema, i que el modificarem per tal de reduir el nombre de setmanes que desarem (de 400 a 10), i activarem la compressió dels logs antics (degut a les limitacions dels nostres discs locals). Per afegir aquests fitxers al servei, ho farem editant el fitxer `/var/411/Files.mk`, i afegint-hi al final el següent:

```
# tail /var/411/Files.mk

FILES += /etc/aliases
FILES += /etc/logrotate.conf
FILES += /etc/profile.d/local.sh
```

Llibreries compartides

Degut a que el nostre objectiu és instal·lar els diferents programes únicament al front-end, en algun moment és possible que sigui necessària alguna llibreria que haguem instal·lat al front-end, però que no estigui als nodes de càlcul. Per tal de combatre això crearem, dins del volum "homes", un directori "`shared_libs`", el qual farà accessible a tots els nodes del clúster aquelles llibreries presents al front-end, però que els nodes

no tinguin instal·lades. Per tal que aquest sistema funcioni, caldrà realitzar diverses tasques:

- Configuració de les llibreries:
  Caldrà modificar el fitxer de configuració de les llibreries de sistema per tal de fer que detecti els nous directoris on estaran les llibreries compartides. Dins del fitxer de configuració `/etc/ld.so.conf` veiem que aquest llegeix la configuració també de tots els fitxers amb extensió "`.conf`" que es trobin dins del directori "`/etc/ld.so.conf.d`". D'aquesta manera, crearem un nou fitxer allà dins el qual inclogui les noves llibreries:

```
# cat /etc/ld.so.conf.d/nfs_shared_libs.conf

/shared_libs/lib
/shared_libs/lib64
/shared_libs/usr/lib
/shared_libs/usr/lib64
/shared_libs/usr/lib/mysql/
/shared_libs/usr/lib64/mysql/
/shared_libs/usr/lib64/atlas/
```

  Addicionalment, modificarem l'ordre de les entrades del fitxer `/etc/ld.so.conf` per tal que les llibreries compartides precedeixin a les locals, de manera que, en cas d'instal·lar una nova versió d'una llibreria, aquesta tingui preferència sobre la versió antiga local.

- Scripts de comprovació i còpia:
  Crearem uns scripts els quals, mitjançant la comanda "`ldconfig -p`" obtindran el llistat de llibreries dels nodes de càlcul, i les compararan amb les presents al front-end. En cas de trobar alguna llibreria "nova" al front-end, aquesta es copiarà al directori de llibreries compartides. Un cop copiades aquestes llibreries, es regenerarà la llista de llibreries de tots els nodes de càlcul mitjançant la comanda "`cluster-fork ldconfig`".

- Actualització de les llibreries:

Per tal de mantenir les llibreries contínuament actualitzades a tots els nodes, afegirem els scripts que hem creat al crontab del front-end, de manera que, cada hora, comprovin si hi ha noves llibreries instal·lades, i en cas afirmatiu les copiï i propagui els canvis pels nodes de càlcul.

- Còpia dels fitxers de configuració als nodes de càlcul:
  Degut a que els fitxers de configuració seran diferents al front-end i als nodes de càlcul (els nodes de càlcul necessitaran carregar les llibreries del directori compartit mentre que al front-end no caldrà), no podrem utilitzar el sistema 411.d per a mantenir actualitzats els fitxers (requereix que els fitxers siguin idèntics i estiguin idènticament situats a tots els nodes). Per a solucionar això, i aprofitant que aquests fitxers no caldrà modificar-los habitualment, optarem per a copiar-los als nodes de càlcul únicament durant la instal·lació del sistema operatiu de cada node de càlcul, en la fase de post-instal·lació.

Configuració dels nodes de càlcul

La configuració de rocks-cluster depèn del graf de kickstart. Aquest, és un graf que defineix la relació que hi ha entre els diferents paquets que composaran el sistema, permetent, entre d'altres coses, definir "tipus d'instàncies" (tipus de hosts) i vincular per a que s'hi instal·lin únicament aquells elements necessaris per al correcte funcionament del sistema.

Pel que respecta al nostre clúster, aquest graph presenta 4 nodes principals: base, client i server:

- Base:

Conté tots els paquets i elements imprescindibles per al funcionament bàsic de qualsevol node. Això inclou paquets com el kernel del sistema, la configuració i particionat de dics, el gestor d'arrencada grub, logs de sistema, configuració de les interfícies ethernet, el servei de firewall, el serveis ssh i ssl, i la instal·lació de llenguatges de scripting com perl, python i tcl.

- Client:

Conté tots aquells paquets necessaris per a configurar un node amb el client dels diversos serveis que correran al clúster. Entre ells estaran els clients de: pxeboot, ssh, 411, autofs, ganglia, ...

- Server:

Conté tots aquells paquets necessaris per a configurar a un node com a servidor del clúster, com és el cas del node que actuarà com a front-end. Això inclou serveis com el web, mysql, els binaris i scripts de control de rocks, nfs, 411, ganglia, routing, pxe, el servidor d'imatges per a l'instal·lador,...

- Hpc:

Composat per diversos sub-paquets, inclou diverses eines per a la computació d'alt rendiment, entre elles mpi, openmpi, i els paquets de base, client i servidor del sistema gestor de cues Sun Grid Engine.

Pel que fa a les instàncies de nodes, el sistema de base ofereix 4 o 5 tipus diferents (que són els que podem escollir quan inserim al sistema un nou node utilitzant la comanda "`insert-ethers`"), però per a l'ús que nosaltres en farem, només dues són rellevants: Compute i Server:

- Compute:

Quan designem un node com una instància d'aquest tipus, el node utilitzarà i instal·larà tots els paquets continguts dins del node client i dins del node base, així com els nodes hpc-client, sge-client i sge-base

- Front-end:

Els hosts d'aquest tipus instal·laran i configuraran tots els paquets continguts dins dels nodes server i base, així com també els continguts per hpc-server, sge-base, sge-server i sge-ganglia.

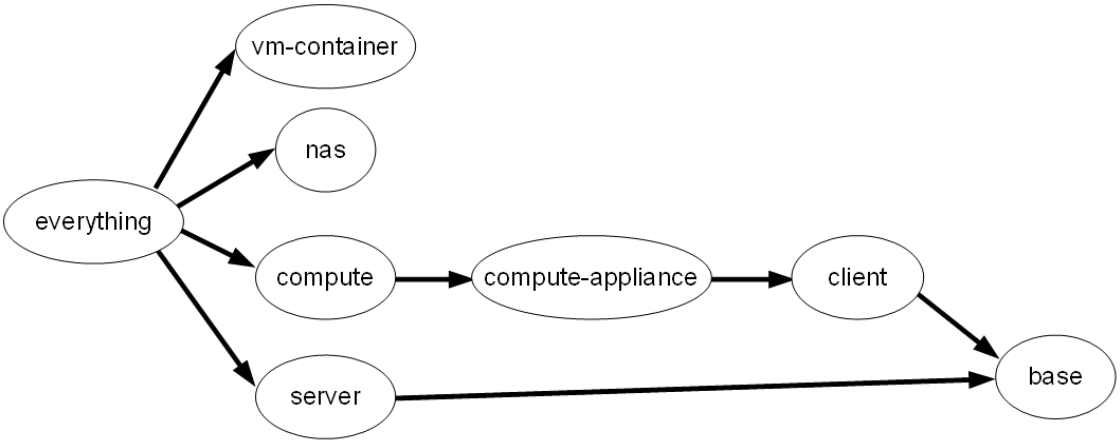

Figura 9.1 - Graf dels tipus d'instàncies. Només es mostren els primers elements del graf.

Podem observar (*figura 9.1*) la relació d'alguns d'aquests paquets. Degut a les dimensions del graf, resulta impossible reproduir-lo adequadament en paper, però el graf complet pot ser consultat aquí: http://bhusers.upf.edu/~theredia/pfc/kickstart-graph.jpg

Personalització i configuració de la imatge dels nodes de càlcul

Per tal d'adaptar el sistema dels nodes de càlcul a les nostres necessitats, caldrà que en modifiquem la imatge amb la qual s'instal·laran. Tant la forma com els nodes del graph de kickstart es troben definits per una multitud de fitxers xml dins del directori `/home/install/rocks-dist/lan/x86_64/build/`.

Per tal de modificar la imatge però, no caldrà que modifiquem aquests fitxers "originals", sinó que rocks permet la modificació d'aquests nodes mitjançant el directori `/home/install/site-profiles/5.0/` dins del qual podrem crear-hi nous fitxers xml els quals substituiran o estendran les funcionalitats del node en qüestió (depenent si anomenem al fitxer extend-node.xml o replace-node.xml). Aquests fitxers xml estan

dividits en 3 apartats: la pre-instal·lació (comandes a executar abans de la instal·lació), la instal·lació (llistat de paquets rpm i comandes a instal·lar) i la post-instal·lació (executada després de la instal·lació del node).

Per a la nostra configuració, necessitarem actualitzar la instal·lació del paquet tcl-devel per una versió més nova (afegint l'rpm a la fase d'instal·lació al fitxer extend-tcl-development.xml) i modificar el particionat de disc (mitjançant el fitxer replace-partition.xml que substituirà completament el node "`partition.xml`" actual). El gruix dels nostres canvis, però, el situarem als apartats d'instal·lació i post-instal·lació del node `extend-compute.xml`. A l'apartat d'instal·lació hi afegirem tots aquells paquets rpm que vagi fent falta instal·lar durant l'ús del sistema, mentre que al de post-instal·lació hi executarem les comandes necessàries per a preparar l'estructura de disc (links a directoris nfs), la configuració personalitzada del correu i dels usuaris ldap, i, principalment, la còpia de tots aquells fitxers de configuració que no sigui possible copiar utilitzant el daemon 411.d (principalment degut a que o no existeixen al front-end, o requereixen configuracions diferents als nodes de càlcul), de manera que ens assegurarem que s'executin un cop finalitzada la instal·lació del sistema (paquets client, base i sge).

La configuració final dels fitxers serà la següent:

- `extend-tcl-development.xml`:

```
<?xml version="1.0" standalone="no"?>
<kickstart>
     <main>
     </main>

     <package>tcl-devel</package>

     <post>
     </post>
</kickstart>
```

- `replace-partition.xml`:

```
<?xml version="1.0" standalone="no"?>
<kickstart>

     <main>
     </main>

     <pre>
        echo "clearpart --all --initlabel --drives=sda
        part / --size 4000 --ondisk sda
        part /var --size 4000 --ondisk sda
        part swap --size 9000 --ondisk sda
        part /state/partition1 --size 1 --grow \
              --ondisk sda" > /tmp/user_partition_info
</pre>

<post>
</post>

</kickstart>
```

- extend-compute.xml:

```
<?xml version="1.0" standalone="no"?>
<kickstart>
     <main>
     </main>
          <!-- SISTEMA -->
     <package>flex</package>
     <package>gsl</package>
     <package>vim-common</package>
     <package>vim-minimal</package>
     <package>vim-enhanced</package>
     <package>java-1.6.0-openjdk</package>
     <package>java-1.6.0-openjdk-devel</package>
     <package>curl</package>
     <package>libidn-devel</package>
     <package>curl-devel</package>
     <!-- /SISTEMA -->

     <!-- PROGRAMES -->
     <package>dejavu-lgc-fonts</package>
     <!-- /PROGRAMES →

     <post>
     <!-- USUARIS LDAP -->
     /usr/sbin/authconfig --enableldap --enableldapauth
          --ldapserver=bhsrv2.s.upf.edu,bhsrv1.s.upf.edu
               --ldapbasedn="dc=upf,dc=edu" --update
```

```
        <!-- LINKS ESTRUCURA DISC -->
    ln -s /home/homes/ /homes
    ln -s /home/scratch/ /scratch
    ln -s /home/backup_desk/ /backup_desk
    ln -s /home/homes/aplic/ /aplic
    ln -s /home/homes/shared_libs/ /shared_libs
    ln -s /state/partition1 /local_disk

    <!-- CONFIGURACIO POSTFIX -->
    /usr/sbin/postconf -e relayhost=bhfront.local \
                  mydomain=local

    <!-- MUNTATGE UNITAT NFS PER A COPIAR FITXERS \
             DE SISTEMA -->
    /bin/mount -t nfs -o nolock,udp  \
             bhfront.local:/export/home/install /mnt
    DIR=/mnt/local/etc

    <!-- LLIBRERIES COMPARTIDES -->
    ln -s /etc/init.d/node-local /etc/ \
                  rc3.d/S99node-local
    ln -s /etc/init.d/node-local /etc/ \
                  rc3.d/K01node-local
    cp ${DIR}/node-local /etc/init.d/

    cp ${DIR}/ld.so.conf /etc/
    cp ${DIR}/nfs_shared_libs.conf /etc/ld.so.conf.d/

    <!-- ANULAR PRECEDENCIA DE LA INSTALACIO \
         JAVA 1.5 PER DEFECTE (package ROCKS-JAVA) \
         I USAR PER DEFECTE JAVA 1.6 -->
    rm -f /etc/profile.d/java.sh
    rm -f /etc/profile.d/java.csh
    ln -s /etc/alternatives/java /usr/bin/java

    <!-- PERFIL DE TOT EL SISTEMA: ALIASES, \
             VARIABLES D'ENTORN,... -->
    cp ${DIR}/local.sh /etc/profile.d/

    <!-- SCRIPT MODIFICAT DE SGEEXECD PER A \
             LLEGIR LES VARIABLES D'ENTORN -->
    cp ${DIR}/sgeexecd /opt/gridengine/default/ \
             common/sgeexecd

    </post>
</kickstart>
```

Finalment, un cop modificats aquests fitxers, només caldrà regenerar la imatge dels nodes utilitzant la comanda "`rocks-dist dist`", i forçar la reinstal·lació dels nodes amb la comanda `/boot/kickstart/cluster-kickstart`. Un cop finalitzada la instal·lació, ja podrem començar a treballar.

Nodes híbrids

En algun moment del procés de disseny i instal·lació del sistema, vam plantejar-nos la incorporació al clúster de càlcul d'un grup de blades que ja teníem al departament, però que estaven infrautilitzats. Aquests nodes són antics, només tenen 2 cores i 2GB de memòria RAM, i, sobretot, cal que segueixin mantenint les seves funcionalitats actuals en cas que siguin necessàries. Per això, caldria crear un nou tipus d'instància de node al graf de kickstart que generi una imatge de disc que tingui en compte tots els detalls i peculiaritats d'aquest tipus de nodes.

Després d'un llarg procés de disseny, instal·lació i proves realitzat principalment pel meu company Angel Carreño, vam arribar a la conclusió que les dues funcionalitats que haurien d'implementar aquests nodes eren massa difícils de combinar, i els nostres caps van decidir cancel·lar el projecte degut a que el temps que necessitaríem invertir en ell no compensava el possible rendiment que poguéssim obtenir del sistema.

Gestió del sistema

Arribats a aquest punt, ja tenim completament instal·lat el sistema operatiu del clúster. Només quedarà configurar diversos elements de gestió per a poder donar accés als usuaris i que aquests comencin a utilitzar-lo, així com elements per a controlar l'estat del sistema.

Usuaris

El primer que caldrà fer és crear els usuaris del sistema. La política que seguirem amb els usuaris serà la següent:

- Utilitzar les dues categories taxonòmiques dins de la base de dades LDAP per a separar usuaris i grups, `ou=Users` i `ou=Groups`.
- Cada usuari tindrà el seu propi grup, el qual serà el seu grup principal, el gid del qual serà el mateix que l'uid de l'usuari. Tant l'id dels usuaris com el dels grups, començaran a partir del número 1001.
- Tots els usuaris pertanyeran també al grup ibe amb gid=20000.
- En cas que diversos usuaris hagin de treballar en un mateix projecte, es crearà un grup per al projecte, s'hi assignaran els usuaris, i es crearà un subdirectori dins del volum scratch per al projecte on hi podran compartir dades.
- En cas que diversos usuaris hagin de compartir dades de manera puntual, habilitarem un subdirectori "`tmp`" al volum scratch, el qual pertanyerà al grup ibe i on tots els usuaris hi tindran permisos de lectura i escriptura. Periòdicament un script comprovarà el contingut del directori i eliminarà tots aquells arxius que hi portin més de 10 dies.

L'estructura de directoris de que disposaran els usuaris serà la següent:

- `/homes/users/USUARI`:

  Home de l'usuari, limitat per quotes i del que es farà backup. Conté enllaços a el scratch i backup_desk de l'usuari, així com a tots els directoris de grups compartits als que pertanyi l'usuari.

  - `public_html`:

    Directori on s'hostatjarà la web personal de l'usuari.
- `/scratch/USUARI`:

  Directori de scratch de l'usuari on emmagatzemar-hi dades grans.
- `/backup_desk/USUARI`:

  Directori on desar el backup del PC de l'usuari, limitat per quotes.

Caldrà generar diversos scripts per a administrar tot això. Concretament caldrà gestionar el següent:

- Altes d'usuaris.
- Baixes d'usuaris.
- Altes de grups.
- Baixes de grups.
- Insercions i delecions d'usuaris dels grups.

Addicionalment, com el frontend serà el punt d'accés de tots els usuaris, caldrà que el protegim de "pics" de càrrega provocats (involuntàriament) per algun usuari i que n'impedeixi el normal funcionament per a la resta d'usuaris. Una de les opcions més segures (i dràstiques) per a resoldre això és la instal·lació d'algun software que generi "gàbies" als usuaris que es connectin via ssh, de tal manera que només vegin una part limitada del sistema (aconseguit mitjançant `chroot`) i només puguin fer servir un subconjunt limitat de comandes. Aquesta opció, tot i que molt segura, resulta completament inviable en el nostre cas, ja que necessitarem que l'usuari sigui conscient de l'estructura del sistema de fitxers per a poder enviar els seus jobs, i també caldrà que pugui executar una gran varietat de comandes i programes per tal de permetre'ls manegar les seves dades, així com testejar scripts o nous programes. Degut a això, optarem per una solució més "lliure", que consistirà només en limitar el nombre de processos, temps d'execució i memòria virtual que podrà utilitzar un mateix usuari. Com aquesta configuració ens interessa que només s'utilitzi al frontend (a la resta de nodes de càlcul se n'encarregarà el SGE), editarem el fitxer `/etc/security/limits.conf` el qual només es carregarà al node frontend, deixant-lo així:

```
# cat /etc/security/limits.conf

*               hard    nproc           2048
*               hard    cpu             180
*               hard    rss             2097152
*               hard    as              2097152

root            hard    nproc           96256
root            hard    cpu             unlimited
root            hard    rss             unlimited
root            hard    as              unlimited
```

De tal manera que els límits del sistema passaran a quedar així:

```
ABANS:
$ ulimit -a
...
cpu time                (seconds, -t) unlimited
max user processes              (-u) 96256
virtual memory           (kbytes, -v) unlimited


Després:
$ ulimit -a
...
cpu time                (seconds, -t) 10800
max user processes              (-u) 2048
virtual memory           (kbytes, -v) 2097152
```

Configuració del sistema gestor de cues

Al finalitzar la instal·lació dels nodes, el sistema gestor de cues SGE (i per tant, el funcionament del clúster de càlcul) ja és plenament funcional, tot i que caldrà modificar la configuració per a millorar el sistema. SGE posa a la nostra disposició multitud d'opcions de configuració, des de la creació de cues, gestió de càrrega, tiquets de consum de recursos, distribució per usuaris, projectes i departaments, paràmetres complexes, configuracions individuals per cada node de càlcul, ... Com inicialment, tant nosaltres com els usuaris, desconeixem quina configuració s'adaptarà millor a les necessitats dels *jobs* que executaran els usuaris, i només podem especular, optarem per a posar el sistema en producció i adaptarem la configuració en diverses iteracions a mesura que vagi evolucionant l'ús del sistema que en fan els usuaris.

La configuració inicial de SGE consisteix en una única cua anomenada "all.q" que conté a tots els nodes de càlcul. Cada node podrà admetre fins a 8 jobs (un per core), i els usuaris es repartiran els jobs utilitzant una cua FIFO. L'única restricció que farà

aquesta configuració bàsica és intentar equilibrar la càrrega dels nodes, distribuint els nous processos sempre a les màquines amb menor càrrega de CPU.

Control del sistema

Un cop tenim preparat el sistema per a l'accés dels usuaris i l'execució de jobs, només caldrà crear i configurar els diversos elements de control per a monitoritzar-ne l'estat. A part dels més típics que ja inclou el sistema, els principals que utilitzarem seran els següents:

Ganglia

El software de control Ganglia ve ja preinstal·lat amb la distribució de Rocks-cluster. Es composa de dos daemons: Gmond, el qual recopila informació sobre l'estat del node i l'envia a través d'un canal multicast, i Gmetad, el qual escolta les comunicacions del canal multicast i en recopila la informació en format xml per tal de fer-la accessible per la interfície web que inclou ganglia.

Figura 9.2 – Panell de control de Ganglia.

La instal·lació inicial per defecte és plenament funcional, però només inclou els nodes del clúster de càlcul (front-end + 8 computes) per lo qual instal·larem manualment el

servei als dos nodes que composen el clúster d'alta disponibilitat. Com ja tenim al front-end actuant com a node central, només caldrà instal·lar als nodes els scripts recol·lectors de dades i configurar-hi el daemon Gmond per a que es comuniqui amb el front-end a través del canal multicast. Un cop fet això només caldrà configurar el Gmetad del front-end per a que escolti els dos nous nodes i els agrupi en un nou conjunt "cluster-HA", i la interfície web ja mostrarà les dades dels nous nodes.

```
cat /etc/gmetad.conf

# The Gmetad configuration file for this cluster.
# Generated automatically by ganglia-server.xml.

data_source "Cluster Unitat Evolutiva" localhost:8649
data_source "Cluster HA" 193.145.57.221:8649 \
          193.145.57.222:8649
```

Processos

Un problema que presenta SGE es que és molt conservador, i en cas que el front-end no pugui comunicar-se amb els daemons que executa als nodes de càlcul, ho interpreta com un "problema temporal de comunicació" i no pren cap decisió. Això implica que quan un node cau de manera inesperada (per exemple, un error de hardware), l'administrador de les cues considera que els jobs encara hi estan corrent, tot i que realment no hi hagi res al node. Per tal de resoldre aquestes situacions, afegirem una entrada al crontab que cridi a un scipt el qual comprovarà les divergències entre el que creu el front-end que hi ha corrent en un node i el que realment hi ha. Degut als possibles falsos positius que es poden provocar (jobs que acaben just en l'instant que s'executa el script, pics de càrrega que fan que la màquina no respongui temporalment, etc.), i a que aquests esdeveniments no seran habituals, decidim que aquest script es limitarà únicament a informar via mail d'aquest successos, requerint intervenció humana per a resoldre'ls, ja que en la majoria d'ocasions que es detecti un problema d'aquest tipus serà necessària igualment (especialment en el cas de problemes de hardware).

Nagios

Tot i que ja tenim Ganglia com a software de monitorització, aquest només ens mostra l'estat de funcionament del sistema, però no ens alerta dels problemes que poden sorgir. Per a cobrir això, al departament ja tenim instal·lat un servidor Nagios. Nagios és un software de control i alerta de servidors que es composa de dos elements: un servidor (que inclou interfície web) i un client (composat per un daemon d'escolta del protocol nrpe i un conjunt de scripts de control, un per cada paràmetre que controlem).

El funcionament és el següent: el servidor disposa d'una llistat del hosts i serveis que ha de controlar a cadascun. Cada cert interval de temps, envia al node una petició d'execució d'un determinat script de control (que el node ja coneix i té accessible) amb els paràmetres adequats en cas de requerir-los. Al client, el daemon "nrpe" escolta l'arribada de la petició, executa el script corresponent i, un cop executat, retorna al servidor un missatge amb l'estat del paràmetre consultat (ok, warning o crític), i aquest últim envia, si cal, un mail d'avís de l'estat problemàtic.

Com al departament ja disposem d'un host que actua com a servidor de Nagios, només caldrà que instal·lem i configurem el client tant als nodes del clúster d'alta disponibilitat com al front-end del clúster de càlcul. Per a fer-ho, només caldrà instal·lar el paquet corresponent a la distribució del host (el clúster d'alta disponibilitat usa SuSE mentre que el de càlcul utilitza CentOS), crear-hi els scripts necessaris (per a controlar aquells paràmetres pels quals els scripts per defecte de Nagios no són adequats), i configurar al fitxer `/etc/nagios/nrpe.cfg` els valors dels llindars de ok/warning/critical dels diferents paràmetres.

- `/etc/nagios/nrpe.cfg`

```
pid_file=/var/run/nrpe.pid
server_port=5666
nrpe_user=nagios
nrpe_group=nagios
allowed_hosts=172.22.202.11
dont_blame_nrpe=1
debug=0
command_timeout=60
```

```
command[check_users]=/usr/lib64/nagios/plugins/check_users -w 5 -c 10
command[check_load]=/usr/lib64/nagios/plugins/check_load -w 15,10,5 -c
30,25,20
command[check_disk1]=/usr/lib64/nagios/plugins/check_disk -w 20 -c 10 -p
/dev/hda1
command[check_disk2]=/usr/lib64/nagios/plugins/check_disk -w 20 -c 10 -p
/dev/hdb1
command[check_root]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w 15% -c
10% -p /
command[check_var]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w 15% -c
10% -p /var
command[check_boot]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w 15% -c
10% -p /boot
command[check_state_partition1]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w 15% -c
10% -p /state/partition1
command[check_rrds]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w 15% -c
10% -p /var/lib/ganglia/rrds
command[check_swap]=/usr/lib64/nagios/plugins/check_swap -w 95% -c 90%
command[check_connections]=sudo
/usr/lib64/nagios/plugins/check_connections 100 200
command[check_date_time]=/usr/lib64/nagios/plugins/check_ntp -H
ntp.upf.edu -w 0.25 -c 1
command[check_quotas]=/usr/lib64/nagios/plugins/check_quotas
command[check_firewall]=/usr/lib64/nagios/plugins/check_firewall
command[check_gmond]=sudo
/usr/lib64/nagios/plugins/check_OS_service_status gmond
command[check_greceptor]=sudo
/usr/lib64/nagios/plugins/check_OS_service_status greceptor
command[check_gmetad]=sudo
/usr/lib64/nagios/plugins/check_OS_service_status gmetad
command[check_sge_qmaster]=/usr/lib64/nagios/plugins/check_ps_process
sge_qmaster
command[check_sge_schedd]=/usr/lib64/nagios/plugins/check_ps_process
sge_schedd
command[check_primary_ldap]=/usr/lib64/nagios/plugins/check_ldap -H
bhsrv2.upf.edu -b "dc=upf,dc=edu"
command[check_secondary_ldap]=/usr/lib64/nagios/plugins/check_ldap -H
bhsrv1.upf.edu -b "dc=upf,dc=edu"
command[check_ldap]=/usr/lib64/nagios/plugins/check_ldap_user acarreno
command[check_running_procs]=/usr/lib64/nagios/plugins/check_procs_Angel R
10 15
command[check_zombie_procs]=/usr/lib64/nagios/plugins/check_procs_Angel Z
5 10
command[check_dead_procs]=/usr/lib64/nagios/plugins/check_procs_Angel X 1
5
command[check_total_procs]=/usr/lib64/nagios/plugins/check_procs_Angel ALL
300 500
command[check_mail]=sudo /usr/lib64/nagios/plugins/check_OS_service_status
postfix
command[check_mail_queue]=sudo /usr/lib64/nagios/plugins/check_mailq -M
postfix -w 1 -c 10
command[check_mem]=/usr/lib64/nagios/plugins/check_mem -f -w 20 -c 10
command[check_subnet_205]=/usr/lib64/nagios/plugins/check_subnet 205
command[check_NFS_volume]=php
/usr/lib64/nagios/plugins/check_disk_and_write_permission.php -w $ARG1$ -c
$ARG2$ -p $ARG3$
command[check_NFS_readonly_volume]=/usr/lib64/nagios/plugins/check_disk -w
$ARG1$ -c $ARG2$ -p $ARG3$
```

Pel que fa a la instal·lació als nodes de càlcul, caldrà que afegim tota la instal·lació al graph de kickstart. Per a fer-ho, caldrà que descarreguem tots els paquets RPM necessaris i els desem a "`/home/install/contrib/5.0/x86_64/RPMS/`"; copiarem els scripts "personalitzats" de nagios i els fitxers de configuració a "`/home/install/local/etc/nagios/`" (no podem replicar-los amb el daemon 411.d ja que alguns dels scripts i paràmetres de configuració seran diferents entre frontend i computes), i, finalment, modificarem el fitxer del graf de kickstart "`/home/install/site-profiles/5.0/nodes/extend-compute.xml`" afegint-hi la instal·lació dels paquets abans esmentats i la còpia dels scripts i fitxers de configuració als seus corresponents directoris, així com la inicialització automàtica del servei d'escolta nrpe. Un cop fet això, regenerem la imatge del sistema amb la comanda "`rocks-dist dist`" i la propera vegada que es reiniciïn els nodes de càlcul ja tindran instal·lat automàticament el servei.

- `extend-compute.xml`:

```
<!-- NAGIOS -->
<package>fping</package>
<package>perl-Crypt-DES</package>
<package>perl-Digest-HMAC</package>
<package>perl-Digest-SHA1</package>
<package>perl-Net-SNMP</package>
<package>nagios-plugins</package>
<package>nagios-nrpe</package>
<!-- /NAGIOS -->

<post>
nagios_plugins_dir=/usr/lib64/nagios/plugins

chkconfig --level 35 nrpe on
cp ${DIR}/nagios/nrpe /etc/init.d/nrpe
cp ${DIR}/nagios/nrpe.cfg /etc/nagios/nrpe.cfg
cp ${DIR}/nagios/check_disk_and_write_permission.php \
          ${nagios_plugins_dir}/
cp ${DIR}/nagios/check_write_permission \
          ${nagios_plugins_dir}/
cp ${DIR}/nagios/check_mem ${nagios_plugins_dir}/
cp ${DIR}/nagios/check_OS_service_status  \
          ${nagios_plugins_dir}/
```

```
cp ${DIR}/nagios/check_firewall ${nagios_plugins_dir}/
cp ${DIR}/nagios/check_connections
${nagios_plugins_dir}/
cp ${DIR}/nagios/check_procs_Angel
${nagios_plugins_dir}/
cp ${DIR}/nagios/check_load_Angel
${nagios_plugins_dir}/
cp ${DIR}/nagios/check_ps_process
${nagios_plugins_dir}/
cp ${DIR}/nagios/check_quotas ${nagios_plugins_dir}/
cp ${DIR}/nagios/check_ldap_user ${nagios_plugins_dir}/
cp ${DIR}/nagios/sge_shepherd_check.php \
          ${nagios_plugins_dir}/
</post>
```

Memòria

Ganglia ens monitoritza l'estat de tots els nodes del sistema de manera excel·lent, però té un problema: passada la hora actual, les gràfiques perden resolució ja que elimina valors per a estalviar espai. Degut a que en ocasions caldrà fer profiling dels jobs executats i necessitarem conèixer el seu comportament i els seus pics de memòria, caldrà que creem una eina per a poder-ho veure. Per a assolir aquesta fita, afegirem al crontab la crida a un script que cada minut obtindrà l'ús de memòria de tots els jobs que estiguin actius al clúster, i desarà aquesta informació en un fitxer de text pla per a cada compute, el creixement dels quals es controlarà mensualment. Per a poder visualitzar aquestes dades, crearem una pàgina web en php que ens permetrà dibuixar les gràfiques del període de temps desitjat (*figura 9.3*).

Scripts

Per a la gestió del sistema caldrà crear diversos scripts que automatitzin certes tasques. Els podem classificar en tres grans grups

- Gestió d'usuaris LDAP:

  Scripts per a facilitar les altes i baixes d'usuaris i grups a la base de dades ldap.

  - Crear usuari.
  - Esborrar usuari.
  - Crear grup i afegir-hi usuaris.
  - Esborrar usuari de grup.

Figura 9.3 – Ús de la memòria del node compute-0-0 durant 4 hores.

- Gestió del sistema de cues:

  Anteriorment hem esmentat la creació de scripts tant per a assegurar l'enviament de tasques a la cua més adequada, com per a penalitzar als usuaris que facin un mal us del sistema de cues.

  - Embolcall qsub.
  - Penalització usuaris.

  Però hem creat també alteres scripts:

  - Epíleg:

    Script d'epíleg que es cridarà cada vegada que acabi un procés, comprovarà l'ús que ha fet de memòria, el registrarà, i, en cas que l'usuari n'hagi fet un mal ús, li enviarà un correu informant-lo i amb instruccions de com evitar-ho en properes ocasions.

  - `qmem`:

    Script que mostra gràficament una instantània de l'estat actual de la memòria i cpu dels nodes i cues del clúster. Complementa a les utilitats incloses al SGE per a visualitzar-ne l'estat (*figura 9.4*). Accessible als usuaris per a que puguin planificar els seus projectes segons l'estat en què es trobi el clúster.

- Control de sistema:

  Necessitarem diversos scripts per a controlar certs aspectes del sistema:

  - Comprovació hosts:

    Control de canvis al fitxer de hosts. A l'executar certes comandes, rocks l'actualitza automàticament. En el nostre cas, com tenim un fitxers diferents al frontend i als computes degut a l'accés a interfícies diferents, caldrà que controlem quan es modifica automàticament i el substituïm per la nostre versió modificada.

  - Comprovació compute caigut:

    Control de discrepàncies entre el que diu el SGE i el que s'està executant realment en un compute degut a caigudes, reinicis o similars.

  - Ús memòria nodes:

    Captura cada minut l'ús de memòria real dels jobs executant-se a cada compute per a poder generar gràfiques.

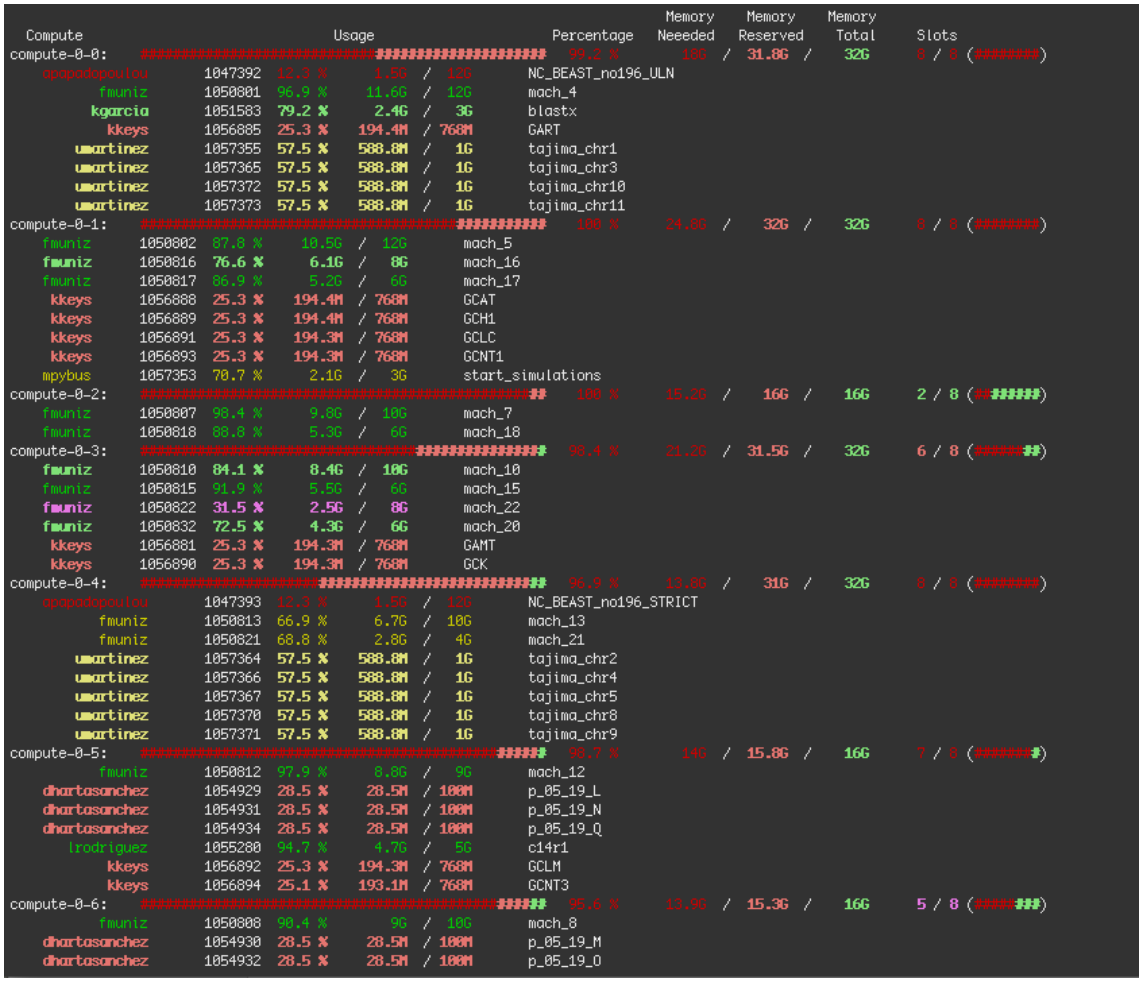

Figura 9.4 – Resultat de la crida a la comanda qmem.

- Retallar log memòria:

  Control de la mida del log anterior, desant només els últims 3 mesos.

- Control llibreries:

  Comprovació periòdica de les llibreries que han sigut instal·lades només al frontend, i copia d'aquestes al disc NFS per a fer-les accessibles als nodes de càlcul.

- Configurar interfície eth1 dels nodes nous:

  La instal·lació inicial d'un nou compute només configura la interfície eth0. Ens caldrà utilitzar aquest script per a configurar-hi la eth1.

Configuració de backup

Donat que aquest servidor contindrà les dades que els usuaris tinguin al clúster de càlcul, caldrà realitzar un backup d'aquestes dades. Aquest backup no serà realitzat per la pròpia màquina, sinó que serà realitzat per un servidor que disposem al grup, dedicat exclusivament a la realització de backups utilitzant el software Tivoli, i connectat al robot de cintes del departament, el qual disposa de capacitat per a 24 cintes de 400GB cadascuna.

Per tal de poder efectuar aquest backup, caldrà però cedir a aquest servidor una partició d'intercanvi de dades, la qual, un cop plena, serà bolcada a les cintes. Per a tal fita hem creat a la cabina de disc el volum “backup_homes”, el qual serà muntat directament via fibra pel servidor de backup (d'aquí la necessitat d'interconnectar la xarxes de fibra nova amb antiga).

La política de backups que seguim és de backups incrementals nocturns, creant inicialment una còpia completa de les dades i a partir d'ella només backups incrementals. Degut a la natura de les dades que utilitzarà el clúster de càlcul, i al gran volum de dades que tindrà respecte el limitat nombre de cintes que disposem (i que serà especialment greu si s'amplia la cabina de disc), limitarem la realització dels backups a les dades contingudes a la partició homes (la qual està limitada per quotes d'usuari), i els fitxers de configuració de sistema que es trobin dins del directori `/etc`.

Annex V, tutorials i documents de suport a l'usuari

Per tal de facilitar la introducció aprenentatge als usuaris del sistema, hem creat diversos tutorials per a ajudar-los, així com pòsters de presentació i la realització de diverses presentacions i seminaris. A continuació mostrarem alguns d'aquests documents:

Web de documentació

Aprofitant que el sistema actuarà com a servidor de webs personals, hem creat una pàgina web que actuarà com a punt d'introducció per als nous usuaris del sistema. Aquesta web estarà disponible a través de l'adreça http://bhusers.upf.edu/~documentation/ i està composada per diverses seccions:

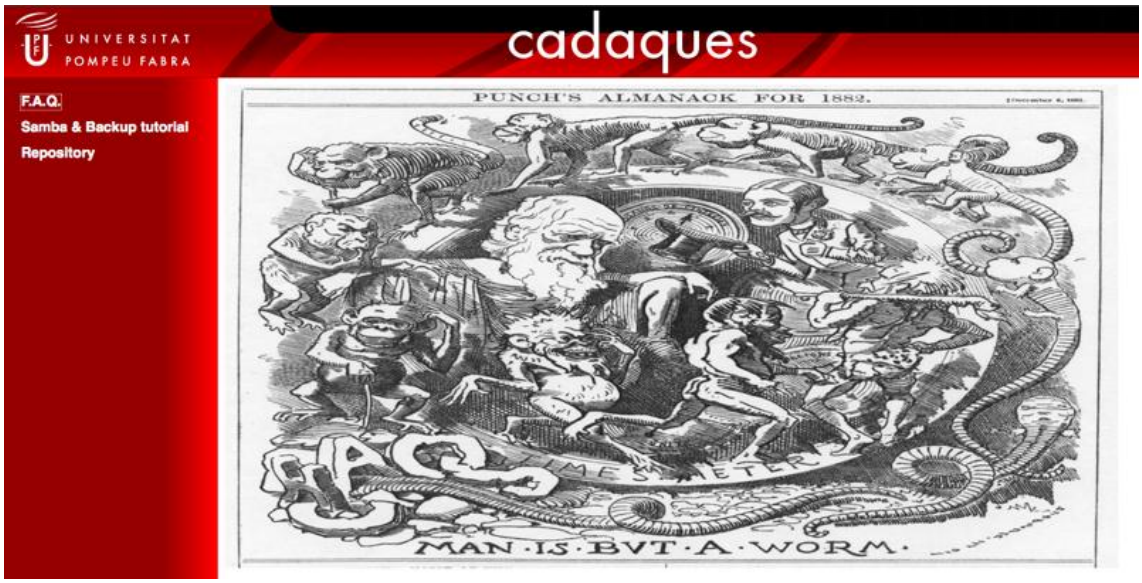

Figura 10.1 – Portada de la web de documentació.

Tutorial

Document d'introducció al sistema per als usuaris. Explica com connectar-se al sistema i dóna unes nocions bàsiques de l'estructura de directoris, enviament de jobs i bones pràctiques d'ús del sistema de cues. A part d'estar disponible aquí, aquest document s'envia adjunt al mail que reben els nous usuaris amb les seves dades d'accés al clúster i tots els usuaris tenen al seu home un link a aquest tutorial per a poder consultar-lo en qualsevol moment.

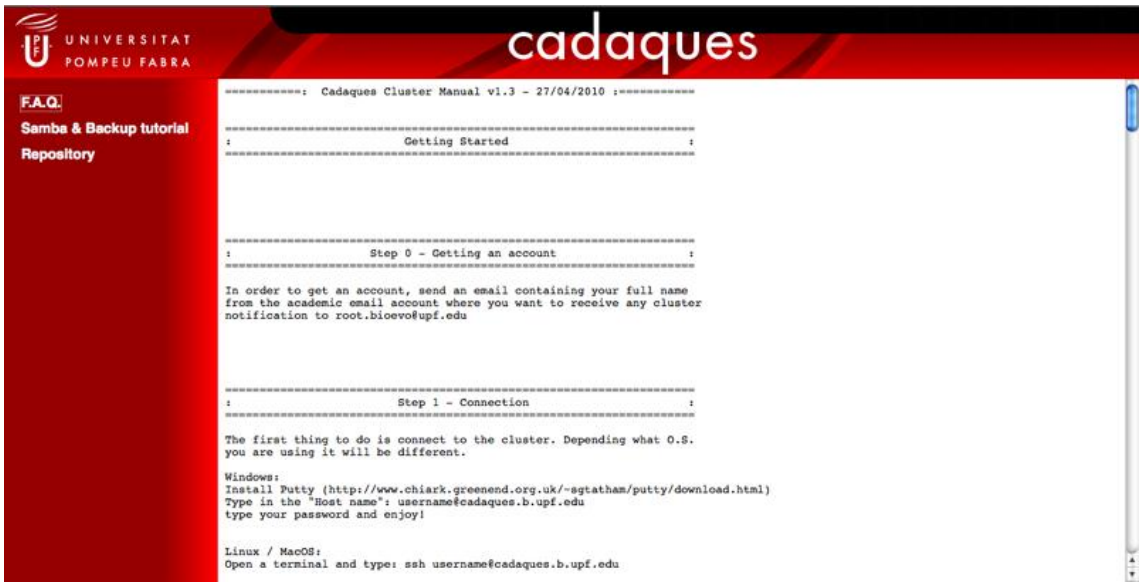

Figura 10.2 – Tutorial d'ús del sistema.

Backup dels PC's de sobretaula

La web inclou també un tutorial sobre com configurar el backup de l'ordinador de l'usuari, amb explicacions pas a pas i detallades per cada sistema operatiu.

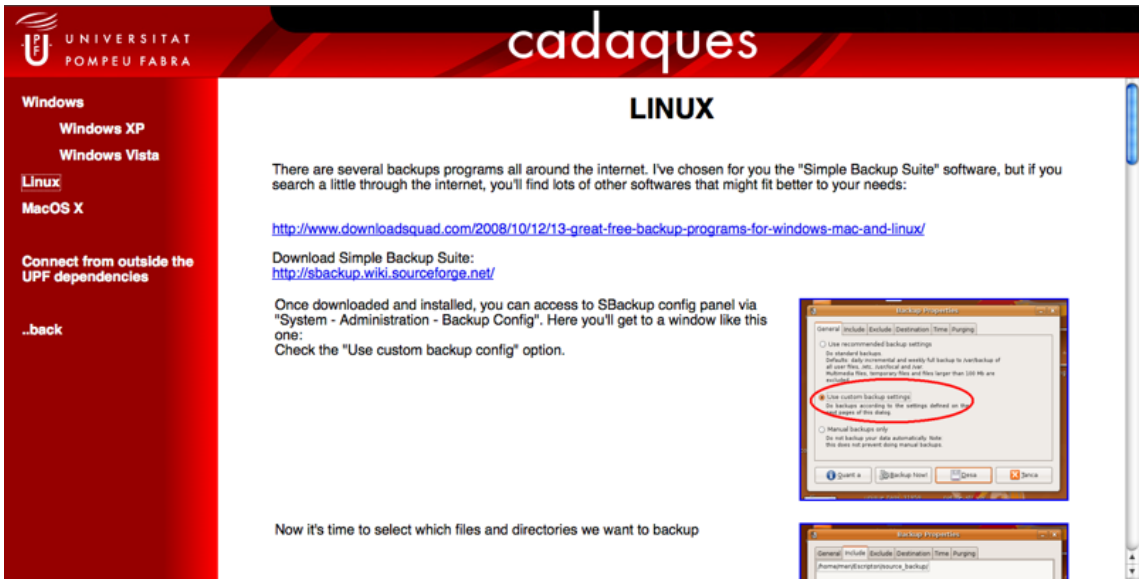

Figura 10.3 – Tutorial pas per pas de configuració del backup de sobretaula.

Connexió externa

Finalment, la web contindrà una breu explicació a com instal·lar la VPN de la universitat per a poder accedir a tot el sistema des de l'exterior de les instal·lacions del Parc de Recerca Biomèdica de Barcelona.

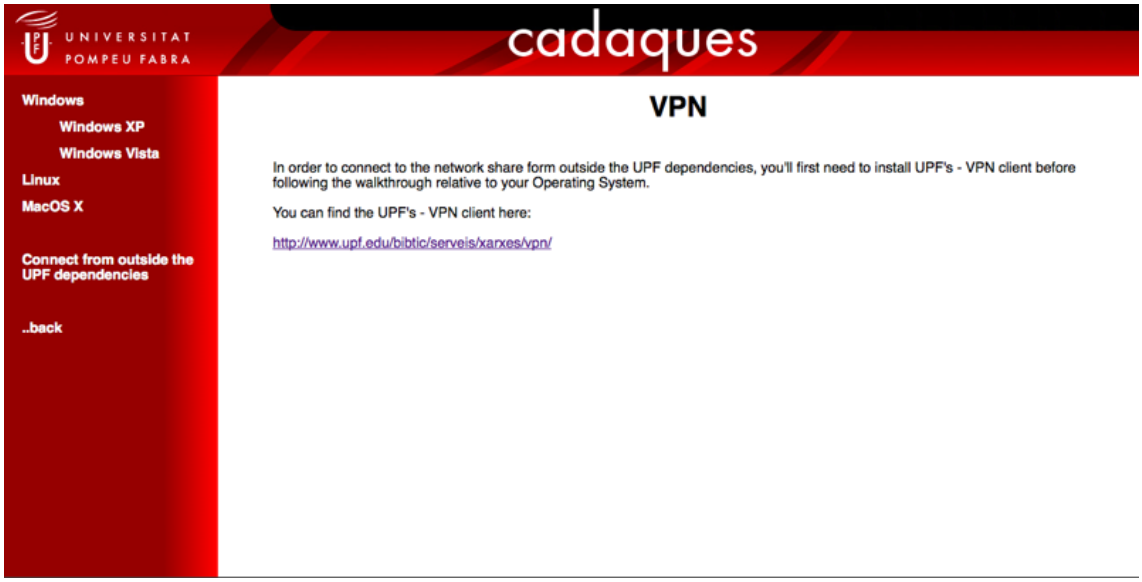

Figura 10.4 – Connexions des de fora al xarxa.

Altres documents introductoris

A part d'aquests tutorials, hem realitzat altres actuacions, seminaris i presentacions, els documents dels quals deixem a continuació:

- Pòster de presentació del clúster al Retreat de l'Institut de Biologia Evolutiva de 2009:
  http://bhusers.upf.edu/~theredia/pfc/poster_retreat_ibe_2009.ppt

- Workshp d'introducció a l'ús de bases de dades genètiques, seminari d'introducció a SQL:
  http://bhusers.upf.edu/~theredia/pfc/seminari_sql.ppt

- Pòster de presentació del clúster al Retreat de l'Institut de Biologia Evolutiva de 2011:
  http://bhusers.upf.edu/~theredia/pfc/poster_retreat_ibe_2011.ppt

- Presentació sobre el present i futur del clúster de càlcul durant el Group Meeting del grup Evolutionary & Primate Genomics:
  http://bhusers.upf.edu/~theredia/pfc/group_meeting_11_2010.ppt

Annex VI, llistat de publicacions que han fet ús del sistema

2008

- Ramírez-Soriano A, Ramos-Onsins SE, Rozas J, Calafell F, Navarro A (2008). *Statistical power analysis of neutrality tests under demographic expansions, contractions and bottlenecks with recombination*. Genetics, 179:555-567. http://www.ncbi.nlm.nih.gov/pubmed/18493071

- Ramírez-Soriano A, Calafell F (2008). *FABSIM: a software for generating FST distributions with various ascertainment biases.* Bioinformatics, 24:2790-2791. http://www.ncbi.nlm.nih.gov/pubmed/18845580

- Comabella M, Craig DW, Camiña-Tato M, Morcillo C, Lopez C, Navarro A, Montalban X and Martin R (2008). *Identification of a Novel Risk Locus for Multiple Sclerosis at 13q31.3 by a Pooled Genome-wide Scan of 500,000 Single Nucleotide Polymorphisms*. PLOS One 3(10):e3490. http://www.ncbi.nlm.nih.gov/pubmed/18941528

- Marquès-Bonet T, Cheng Z, She X, Eichler EE and Navarro A (2008). *The genomic distribution of intraspecific and interspecific divergence of human SDs*. BMC Genomics 9:384. http://www.ncbi.nlm.nih.gov/pubmed/18699995

- Morcillo-Suarez C, Alegre J, Sangros R, Gazave E, de Cid R, Milne R, Amigo J, Ferrer-Admetlla A, Moreno-Estrada A, Gardner M, Casals F, Pérez-Lezaun A, Comas D, Bosch E, Calafell F, Bertranpetit J and Navarro A (2008). *SNP Analysis To Results (SNPator): a web-based environment oriented to statistical genomics analyses upon SNP data*. Bioinformatics 24(14):1643-4. http://www.ncbi.nlm.nih.gov/pubmed/18699995

- Ferrer-Admetlla A, Bosch E, Sikora M, Marquès-Bonet T, Ramírez-Soriano A, Navarro A, Lazarus R, Calafell F, Bertranpetit J and Casals F (2008). *Balancing selection is the main force shaping the evolution of innate immunity genes*. Journal of Immunology. 181:1315-22.
  http://www.ncbi.nlm.nih.gov/pubmed/18606686

2009

- Bosch E, Laayouni H, Morcillo C, Casals F, Moreno-Estrada A, Ferrer-Admetlla A, Gardner M, Rosa A, Navarro A, Comas D, Graffelman J, Calafell F, Bertranpetit J (2009). *Decay of linkage disequilibrium within genes across HGDP-CEPH human samples: most population isolates do not show increased LD*. BMC Genomics 10(1):338.
  http://www.ncbi.nlm.nih.gov/pubmed/19638193

- Ferrer-Admetlla A, Sikora M, Laayouni H, Esteve A, Roubinet F, Blancher A, Calafell F, Bertranpetit J and Casals F (2009). *A natural history of FUT2 polymorphism in humans*. Mol Biol Evol. 2009 Sep;26(9):1993-2003. Epub 2009 Jun 1.
  http://www.ncbi.nlm.nih.gov/pubmed/19487333

- Sikora M, Ferrer-Admetlla A, Laayouni H, Menendez C, Mayor A, Bardaji A, Sigauque B, Mandomando I, Alonso PL, Bertranpetit J and Casals F (2009). *A variant in the gene FUT9 is associated with susceptibility to placental malaria infection*. Hum Mol Genet. 2009 Aug 15;18(16):3136-44. Epub 2009 May 21.
  http://www.ncbi.nlm.nih.gov/pubmed/19460885

- Garagnani P, Laayouni H, González-Neira A, Sikora M, Luiselli D, Bertranpetit J, Calafell F (2009). *Isolated populations as treasure troves in genetic epidemiology: the case of the Basques*. Eur J Hum Genet. 2009 Nov;17(11):1490-4. Epub 2009 May 6.
  http://www.ncbi.nlm.nih.gov/pubmed/19417765

- Casals F, Ferrer-Admetlla A, Sikora M, Ramírez-Soriano A, Marquès-Bonet T, Despiau S, Roubinet F, Calafell F, Bertranpetit J, Blancher A (2009). *Human pseudogenes of the ABO family show a complex evolutionary dynamics and loss of function*. Glycobiology. 2009 Jun;19(6):583-91. Epub 2009 Feb 13.
- http://www.ncbi.nlm.nih.gov/pubmed/19218399

- Moreno-Estrada A, Tang K, Sikora M, Marquès-Bonet T, Casals F, Navarro A, Calafell F, Bertranpetit J, Stoneking M and Bosch E (2009). *Interrogating 11 fast-evolving genes for signatures of recent positive selection in worldwide human populations.* Molecular Biology and Evolution 26(10): 2285- 2297. http://www.ncbi.nlm.nih.gov/pubmed/19578157

- Camiña-Tato M, Morcillo-Suárez C, Navarro A, Fernandez M, Horga A, Oksenberg JR, Montalban X, Comabella M (2009). *Genetic association between polymorphisms in the BTG1 gene and multiple sclerosis*. Journal of Neuroimmunology 213(1-2):142-7. http://www.ncbi.nlm.nih.gov/pubmed/19515430

- Comabella M, Craig DW, Morcillo-Suarez C, Rio J, Navarro A, Fernandez M, Martin R, Montalban X (2009). *Genome-wide Scan of 500,000 Single Nucleotide Polymorphisms in responders and non-responders to interferon-beta in Multiple Sclerosis*. Archives of Neurology 66(8):972-8. http://www.ncbi.nlm.nih.gov/pubmed/19667218

- Marques-Bonet T, Kidd JM, Ventura M, Graves TA, Cheng Z, Hillier LW, Jiang Z, Baker C, Malfavon-Borja R, Fulton LA, Alkan C, Aksay G, Girirajan S, Siswara P, Chen L, Cardone MF, Navarro A, Mardis ER, Wilson RK, Eichler EE (2009). *A Burst of Segmental Duplications in the African Great Ape Ancestor*. Nature 457:877-81. (Published on February the 12th 2009, in the special issue commemorating Darwin's birthday). http://www.ncbi.nlm.nih.gov/pubmed/19212409

- Marsillach J, Aragonès G, Beltrán R, Caballeria J, Pedro-Botet JC, Morcillo-Suárez C, Navarro A, Joven J and Camps J (2009). *The measurement of the lactonase activity of paraoxonase-1 in the clinical evaluation of patients with chronic liver impairment*. Clinical Biochemistry 42(1-2):91-98. http://www.ncbi.nlm.nih.gov/pubmed/18977341

2010

- Melé M, Javed A, Pybus M, Calafell F, Parida L, et al. (2010). *A New Method to Reconstruct Recombination Events at a Genomic Scale.* PLoS Comput Biol 6(11): e1001010. doi:10.1371/journal.pcbi.1001010. http://www.ncbi.nlm.nih.gov/pubmed/21124860

- Camiña-Tato M, Morcillo-Suárez C, Fernandez M, Martin R, Ortega I, Navarro A, Sánchez A, Carmona P, Julià E, Tortola M, Audí L, Fossdal R, Oksenberg JR, Montalban X, Comabella M (2010). *Gender associated differences of perforin polymorphisms in the susceptibility to multiple sclerosis*. The Journal of Immunology 185: 5392 -5404. http://www.ncbi.nlm.nih.gov/pubmed/20921521

- Al-Shahrour F, Minguez P, Marqués-Bonet T, Gazave E, Navarro A and Dopazo J (2010). *Selection upon genome architecture: conservation of functional clusters with changing genes*. PLoS Computational Biology 6(10):e1000953. http://www.ncbi.nlm.nih.gov/pubmed/20949098

- Camiña-Tato M, Fernández M, Morcillo-Suárez C, Navarro A, Julià E, Edo MC, Montalban X, Comabella M (2010). *Genetic association of CASP8 polymorphisms with primary progressive multiple sclerosis*. Journal of Neuroimmunology 222:70-75. http://www.ncbi.nlm.nih.gov/pubmed/20363033

2011

- Sikora M, Laayouni H, Calafell F, Comas D and Bertranpetit J (2011). *A genomic analysis identifies a novel component in the genetic structure of sub-Saharan African populations*. Eur J Hum Genet. 2011 Jan;19(1):84-8. Epub 2010 Aug 25.
  http://www.ncbi.nlm.nih.gov/pubmed/20736976

- Marigorta UM, Lao O, Casals F, Calafell F, Morcillo-Suárez C, Faria R, Bosch E, Serra F, Bertranpetit J, Dopazo H and Navarro A (2011). *Recent human evolution has shaped geographical differences in susceptibility to disease.* BMC Genomics 2011, 12(1):55.
  http://www.ncbi.nlm.nih.gov/pubmed/21261943

- Henn BM, Gignoux CR, Jobin M, Granka JM, Macpherson JM, Kidd JM, Rodríguez-Botigué L, Ramachandran S, Hon L, Brisbin A, Lin AA, Underhill PA, Comas D, Kidd KK, Norman PJ, Parham P, Bustamante CD, Mountain JL, Feldman MW. *Hunter-gatherer genomic diversity suggests a southern African origin for modern humans*. Proc Natl Acad Sci U S A. 2011 Mar 29;108(13):5154-62. Epub 2011 Mar 7.
  http://www.ncbi.nlm.nih.gov/pubmed/21383195

- Laayouni H, Montanucci L, Sikora M, Melé M, Dall'Olio GM, Lorente-Galdos B, McGee KM, Graffelman J, Awadalla P, Bosch E, Comas D, Navarro A, Calafell F, Casals F, Bertranpetit J (2011). *Similarity in recombination rate estimates highly correlates with genetic differentiation in humans*. PLoS One. 2011 Mar 28;6(3):e17913.
  http://www.ncbi.nlm.nih.gov/pubmed/21464928

- Muñiz-Fernandez F, Carreño-Torres A, Morcillo-Suarez C and Navarro A (2011). *Genome-wide Association Studies Pipeline (GWASpi): a desktop application for genome-wide SNP analysis and management*. Bioinformatics (Advance Online Publication).
  http://www.ncbi.nlm.nih.gov/pubmed/21586520

- The Orangutan Genome Consortium (Including Navarro A. and 6 members of my team). 2011. *Comparative and demographic analysis of orang-utan genomes*. Nature 469: 529–533. http://www.ncbi.nlm.nih.gov/pubmed/21270892

- Henn BM, Rodríguez-Botigué L, Gravel S, Wang W, Brisbin A, Fadhlaoui-Zid K, Zalloua PA, Moreno-Estrada A, Bertranpetit J, Bustamante CD, Comas D (2011). *Genomic Patterns Reveal Complex Human Admixture and Migration into North Africa*. Submitted in PLoS Genetics on April 2011.

- Gazave E, Darré F, Morcillo-Suarez C, Petit-Marty N, Carreño-Torres A, Martinez-Marigorta U, Ryder OA, Blancher A, Rocchi M, Bosch E, Baker C, Marquès-Bonet T, Eichler EE and Navarro A (2011). *Copy Number Variation Analysis In The Great Apes Reveals Human-Specific Fixations And Species-Specific Patterns Of Structural Variation*. Genome Research (In press).

- Lorente-Galdos B, Medina I, Morcillo-Suarez C, Sangrós R, Alegre J, Pita G, Vellalta G, Malats N, Pisano DG, Dopazo J and Navarro A (2011). *SYSNPs (Select Your SNPs): a web tool for automatic and massive selection of SNPs*. Int. J. of Data Mining and Bioinformatics (In Press).

En preparació

- Martinez-Cruz B et al. *The Basques in the Iberian genetic landscape: Y-chromosome and mitochondrial DNA revisited*. In prep.

- Martinez-Cruz B et al. *Wendish populations within present and former ethnic German territories show unidirectional Y-chromosomal gene flow from Slavic to German populations and rapid demographic growth before the Slavic expansion in Europe*. In prep.

- Luisi P, Alvárez-Ponce D, Dall'Ollio G, Bertranpetit J, Laayouni H. *Recent positive selection in the human insulin/TOR signal transduction pathway: network-level analysis*. In prep.

- Patillon B, Luisi P, Sabbagh A, Genin E. *Recent positive selection in VKORC1,a gene involved in response to drugs*. In prep.

- Garcia-Garcerà M, Coscollà M, Bonet N, Durbán A, Latorre A, Calafell F. *A new method for prokaryotic enrichment of skin tissue samples allow the metagenomic analyisis of skin*. In prep.

- Garcia-Garcerà M, Bonet N, Coscollà M, Garcia-Etxebarria K, Calafell F. *Long term immunodeficiency is associated with a Selective Shift in Mice Skin microbiota*. In prep.

- Keys K, Montanucci L et al. *Evolutionary and network-level analysis of the 70 metabolic pathways of the EHMN database (EHMN: Edinburgh Human Metabolic Network)*. In progress.

- Dall'Olio GM, Laayouni H, Luisi P, Sikora M, Montanucci L and Bertranpetit J. *The structure of adaptive selection in a metabolic pathway among human populations: the case of Asparagine N-Glycosylation*. In prep.