Outils d'exploitation de grappes de PC

Philippe Augerat, Simon Derr, Stéphane Martin, Céline Robert
Laboratoire ID/IMAG

Résumé : Ce document décrit les différents choix d'exploitation d'une grappe de 225 PC installée à l'INRIA en Avril 2001. Nous présentons tout d'abord le type de grappe installée et son usage. Nous décrivons ensuite les logiciels installés puis la procédure choisie pour l'installation du système d'exploitation ce qui constitue l'aspect le plus original de l'administration d'un cluster. La dernière partie décrit le problème de passage à l'échelle rencontré lorsque l'on veut exploiter une grappe de plusieurs centaines de machines. Nous conclurons en décrivant les perspectives qu'ouvrent ce travail pour l'exploitation d'un parc de machines de grande taille et de grappes virtuelles construites à partir des machines non utilisées d'un Intranet

INTRODUCTION

Les performances des ordinateurs de bureau standards à base de processeur Intel ont atteint un tel niveau qu'il est aujourd'hui possible de construire à partir de ces machines des supercalculateurs aussi rapides que les calculateurs dédiés type CRAY, IBM SP ou SGI Origin 2000. D'autre part, des problèmes technologiques ou simplement les coûts de certains composants (commutateurs pour l'accès à la mémoire par exemple) limitent les possibilités de passage à l'échelle des machines multiprocesseurs (SMP). Dans ce contexte, les grappes de PC apparaissent comme une très bonne alternative aux autres moyens de calculs.

En revanche, du point de vue de l'utilisateur ou de l'administrateur, ces machines ont l'inconvénient d'apparaître comme un simple empilement de ressources peu différent de l'ajout des machines d'un réseau Intranet par exemple. Le succès des architectures de type grappe passe donc par le développement d'outils permettant de voir (accéder, programmer, administrer) une grappe de centaines de machines comme s'il s'agissait d'une seule machine.

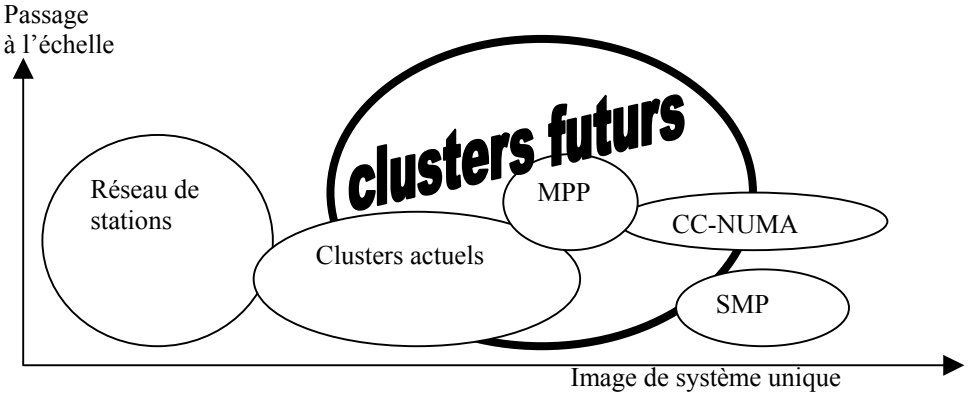

Figure 1 : Les machines pour le calcul parallèle et distribué

Nous appellerons donc grappe (ou cluster en anglais) un ensemble de PC reliés en réseau et considérés comme une ressource unifiée de calcul. Ce document décrit les différents choix d'exploitation d'une grappe de 225 PC et les développements réalisés pour optimiser son usage et son administration. La première partie de l'article va présenter le type de grappe installée et son usage. La seconde partie les services et logiciels installés. La troisième partie détaille la procédure d'installation de la grappe. La dernière partie présente la problématique du passage à l'échelle des outils d'exploitation. Nous conclurons en décrivant les perspectives qu'ouvrent ce travail pour l'exploitation d'un parc de machines de grande taille et de grappes virtuelles construites à partir des machines non utilisées d'un Intranet.

LA GRAPPE

Le matériel

Nous considérerons ici une architecture de grappe banalisée (aussi appelée grappe Beowulf[1]) : 225 PC monoprocesseurs dédiés au calcul hautes performances sont reliés par un réseau Ethernet lui aussi dédié, mixant des vitesses allant de 100 mégabits/s sur l'interface réseau des machines à 1 gigabit/s sur les interfaces des commutateurs.

[1] Grappe générique dont les composants matériels et logiciels sont des standards de l'industrie et les environnements non propriétaires.

Cette grappe est constituée de 225 machines (ou nœuds) sous Linux. Chaque nœud est un PC équipé d'un processeur Pentium III de 256 Mo de mémoire, 15 gigas de disque et d'un port réseau Fast Ethernet. Il s'agit de machines ne disposant ni de slot PCI d'extension, ni de lecteur de disquette. Les 225 machines sont connectées à un réseau Ethernet matérialisé par 5 commutateurs reliés entre eux par des liens en Gigabit.

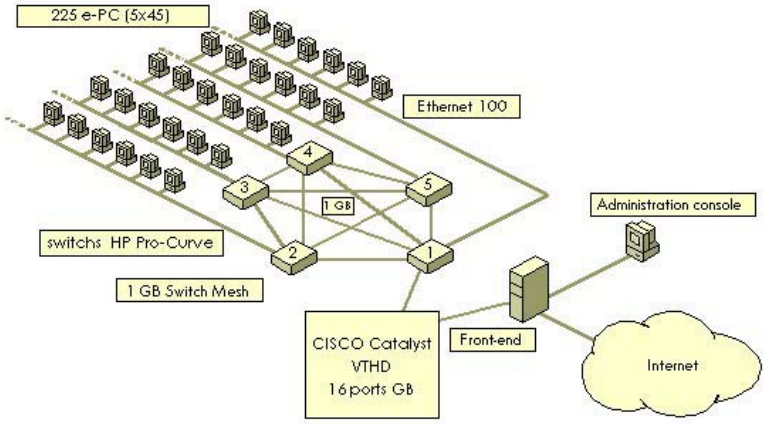

Figure 2 : Architecture du I-Cluster (INRIA / Lab. ID-IMAG)

L'interconnexion des commutateurs (et donc la hiérarchie des communications) est facilement modifiable en fonction de l'utilisation souhaitée de la grappe (anneau, double anneau, arbre, étoile …)
Un des commutateurs est aussi connecté au réseau très haut débit (VTHD[2]) déployé sur tout le territoire français par un lien à 1 Gigabit. Le fond de panier de chaque commutateur est de 3.8 Gigabits ce qui constitue donc aussi à peu près la bande passante de bissection de la grappe.

L'utilisation de la grappe

Si l'usage premier de la grappe est actuellement le calcul scientifique, son acquisition a pour objet à moyen terme un travail de recherche sur le déploiement d'une architecture de services sur l'Intranet d'une entreprise. Ce travail prolonge les recherches du laboratoire ID dans le domaine de la programmation parallèle (outils d'exploitation, dont le débogage, et programmation d'applications) sur les grappes de SMP et les supercalculateurs. Les applications visées couvrent principalement les domaines du calcul scientifique et du traitement de l'information.
Une partie des utilisateurs est aussi intéressée par le couplage de la grappe avec d'autres ressources de calcul (autres grappes de PC, studio de réalité virtuelle, supercalculateurs à mémoire partagée,…) pour aboutir au couplage de gros code applicatifs à travers VTHD.

Le I-Cluster possède aujourd'hui plus de 70 utilisateurs réguliers et a donné lieu à une production scientifique importante sur différentes applications comme la dynamique moléculaire, l'imagerie, la résolution de problèmes NP-difficile d'optimisation combinatoire,

[2] Plate-forme d'expérimentation IP/WDM Vraiment Très Haut Débit pour applications de l'Internet nouvelle génération (France Telecom, Inria, Enst, Enst-br, Int, Institut Eurecom)

l'océanographie, etc.) et sur la recherche en informatique (couplage de codes, parallélisation de serveurs scientifiques, environnement d'exploitation pour grappes de PC, etc.).

Afin de mesurer les performances de cette architecture, un certain nombre de programmes tests ont été exécutés sur la grappe de PC. Le principal benchmark utilisé est Linpack [14], un code de factorisation de matrices denses constitué d'opérations en calculs flottants en double précision et de communications synchrones. Ce benchmark permet en particulier de comparer les supercalculateurs les plus puissants du monde au sein d'un classement baptisé TOP500 [9]. Avec ce même test, il s'agit aussi d'évaluer le passage à l'échelle des performances d'un réseau de PC standard. Les résultats sont extrêmement positifs. D'une part, le I-Cluster réalise 81,6 Gflop/s avec Linpack, une performance qui le positionne au 385ème rang dans le TOP500. D'autre part, les résultats passent à l'échelle de manière quasi-linéaire à la fois en terme de performances et de ratio prix/performance depuis une configuration à 2 machines jusqu'à une configuration à 225 machines.

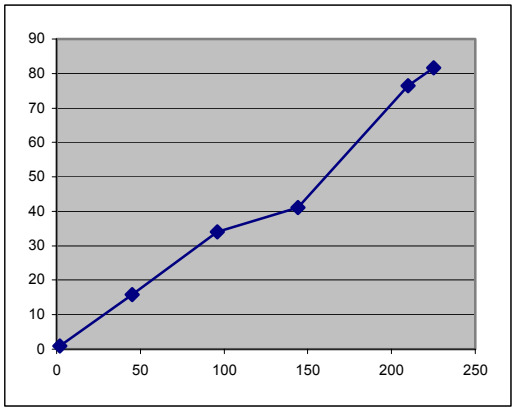

Figure 3 : passage à l'échelle : Gflop/s Linpack de 2 à 225 nœuds

Avec un coût par Gflop/s de 20 kF, le I-Cluster domine bien sûr en terme de prix/performances toute autre solution de type supercalculateur ou grappe haute performance. Ceci est vrai pour toute une gamme d'applications, semblables à Linpack où le rapport entre les temps de calcul et les communications entre machines est relativement grand.

Nous avons testé de manière intensive le comportement des commutateurs notamment lors de communications bipoint (1-1 /n paires).

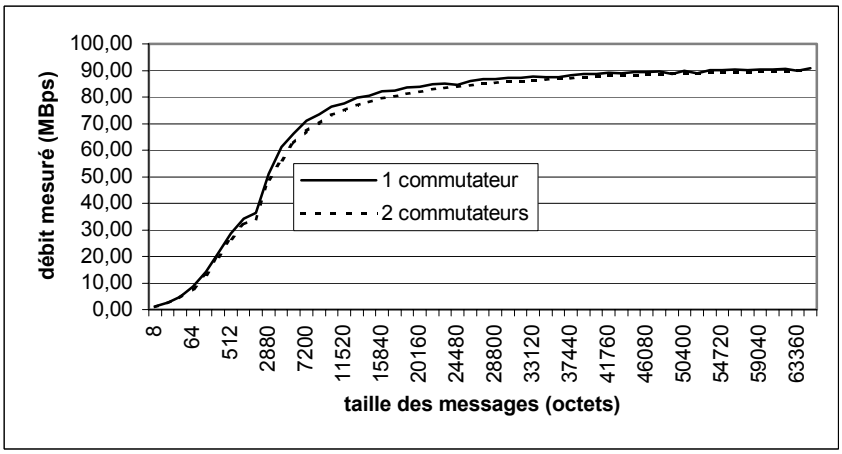

Figure 4: courbe de débit en fonction de la taille du paquet

La figure 4 a été construite à partir de mesures de communications entre paires indépendantes sur un double anneau. Pour un commutateur les mesures sont faites avec 22 paires indépendantes. La courbe pour deux commutateurs concerne 45 paires indépendantes. L'algorithme utilisé est un ping-pong, la taille du paquet augmentant de 0 a 64800 octets.
Les débits observés sont proches de 90 Mbit/s pour une valeur théorique de 100 Mbit/s et les courbes sont identiques à celles observées lors de tests de communications bipoint pour une paire.
Suivant l'architecture étudiée nous observons deux types de latences de communication : soit une latence locale au commutateur (7 micro-secondes) soit une latence indiquant le nombre de commutateurs à traverser (multiple de 7). Par exemple dans le cas d'une architecture en pentagramme, ou chaque machine est a deux commutateurs au plus d'une autre machine, nous observons deux latences différentes : la latence locale et celle correspondant à la traversée d'un second commutateur.

LOGICIELS INSTALLES SUR LE I-CLUSTER

Il est en général possible d'utiliser telle quelle une distribution Linux standard sur un cluster. L'image installée sur les machines du I-Cluster est une distribution Linux Mandrake 7.1 à laquelle on a ajouté certaines librairies ou applications spécifiques du calcul parallèle et des outils d'exploitation. Si on veut utiliser un environnement plus évolué, matériel (réseau haut débit) ou logiciel (par exemple utiliser le logiciel Mosix pour migrer automatiquement des processus entre machines), on devra recompiler le noyau Linux en ajoutant les fonctionnalités correspondantes.

Un certain nombre de distribution Linux pour cluster « clé en main » sont apparues récemment. Il s'agit souvent de la compilation brute de packages (Beowulf, Suse, ACE, etc.) [10] ou de solutions intégrées, souvent payantes, prenant en compte une utilisation précise et limitée du cluster (programmation parallèle par passage de message par exemple) dans un environnement matériel spécifique (Wulfkit de Scali, Raisin de Alinka [2], Scyld [3], etc. [4], [11]) ou enfin de distribution rassemblant les outils d'un centre de recherche (Score). Nous avons fait le choix de créer une solution ad-hoc car aucune des distributions précédentes n'avait été plébiscitée par le monde du calcul parallèle et encore moins validée à l'échelle d'une grappe de plusieurs centaines de PC.

On distingue sur le I-Cluster deux types d'installation : les machines serveurs (2 sur le I-Cluster) et les machines de calculs (225 sur le I-Cluster). L'installation de ces machines est réalisé par un clonage dont on décrit la procédure dans le chapitre déploiement.

Services/administration

Sur les serveurs de la grappe, on retrouve un certain nombre de services classiques sur un parc de machines en particulier des services réseau : authentification (NIS, DHCP, SSF/SSH,…), fichiers (NFS, TFTP,…), gestionnaire de licences, etc. Ces services sont configurés dans le but de minimiser leur intrusion dans les applications. Par exemple, nous utilisons le service Unix standard NIS (Network Information Service) pour l'authentification des utilisateurs mais chaque machine de calcul est configurée comme un serveur NIS esclave de manière à éviter que la centralisation des informations soit source de trafic réseau. Seules les modifications de la base d'information entraînent un trafic réseau. Le cas des performances des systèmes de fichiers est traité plus loin.

L'administration du I-Cluster se fait à l'aide d'extensions « grappe » des commandes Unix standard. Le grand nombre de machines implique que ces extensions soient conçues de manière parallèle ou distribuée. La parallélisation des commandes *rsh* et *rcp* suffit en pratique pour construire un ensemble plus vaste de commandes qui sont utilisées à la fois par les administrateurs et les utilisateurs(*ls*, *ps*, *find*, *kill*, *rm*, etc.).

Environnement de calcul

Les divers langages pour le calcul parallèle sont installés sur le I-Cluster : MPI, PVM, CORBA, JAVA ainsi que des librairies mathématiques optimisées : BLAS/PBLAS, SCALAPACK/LAPACK. Ces logiciels doivent en général être recompilés pour s'adapter aux caractéristiques des machines, du réseau et des compilateurs disponibles.
Des outils facilitent ce travail, en particulier le logiciel ATLAS qui permet de générer des librairies mathématiques adaptées au processeur et à la hiérarchie mémoire (caches) de la machine.

La plupart des compilateurs professionnels sont maintenant disponibles sous Linux. Nous utilisons le compilateur Portland en complément aux compilateurs GNU.

Une interface de système unique pour l'utilisateur

L'exploitation d'une grappe de calcul nécessite l'utilisation d'un outil d'allocation des ressources aux différents utilisateurs ; cet outil doit être complété par des modules d'ordonnancement des taches et de réservations de machines. Plusieurs solutions sont utilisées par les différents centres de calculs (LSF, PBS, Maui, GnuQueue…). Les machines du I-Cluster sont accessibles via openPBS (Portable Batch System) dans sa version 2.3.12 [13].
PBS autorise l'allocation d'un ensemble de machine à une tâche créée par un utilisateur ainsi qu'une gestion des files d'attentes en fonction de paramètres configurables par les administrateurs du système (durée maximale de la réservation, nombre de processeurs voulu, mémoire demandée…). Des scripts exécutés avant et après chaque tâche permettent de maintenir les nœuds dans un état « propre » entre chaque allocation.

PBS constitue un des éléments d'une interface de système unique. Un frontal qui dispatche les utilisateurs sur des nœuds de login et de développement strictement identiques, et la mise à disposition via NFS d'une hiérarchie de fichiers unique sur toutes les machines constituent les autres éléments de cet interface.

DEPLOIEMENT D'UNE GRAPPE DE PC

Vu le nombre de machines, l'installation des PC doit donc se faire par le réseau pour éviter des tâches manuelles répétitives. La taille des données a transférer (plusieurs gigas) impose une exploitation optimale du réseau. L'installation d'une machine vierge oblige a écrire un secteur de boot sur la machine cliente. Pour cela, un serveur fournit une image de système, créée lors de l'installation manuelle d'un premier PC qui servira de serveur d'installation.

Un nœud vierge démarre sur sa carte réseau et charge depuis le réseau en plusieurs étapes les informations et le système d'exploitation qu'il va utiliser. C'est le protocole PXE, implanté dans la bootrom de la carte Ethernet qui est appelé pour interroger des serveurs DHCP et TFTP qui fourniront les programmes et images d'installation.
Cette installation est utilisée classiquement pour installer des postes de travail dans une salle de TP par exemple avec le logiciel BpBatch [6]. BpBatch est un programme capable d'interpréter de petits scripts écrits dans un langage spécifique, donnant ainsi la possibilité de démarrer une machine de plusieurs manières différentes en fonctions de certains paramètres ou de l'intervention d'un utilisateur. BpBatch est notamment capable de partitionner un disque dur, de formater des partitions, de créer des fichiers, faire démarrer la machine sur son disque dur, ou sur une image de disquette, on encore de démarrer un noyau Linux charge a travers le réseau.

Lorsque plusieurs PC doivent être installés en même temps, il est nécessaire d'utiliser une procédure où le serveur d'image ne risque pas la saturation. Plusieurs possibilités de distribution d'images peuvent être utilisées, comme le multicast, la diffusion en arbre en plusieurs étapes, et la solution que nous avons adoptée, la mise en place d'une chaîne de diffusion permettant la copie en une seule étape du système de fichiers.

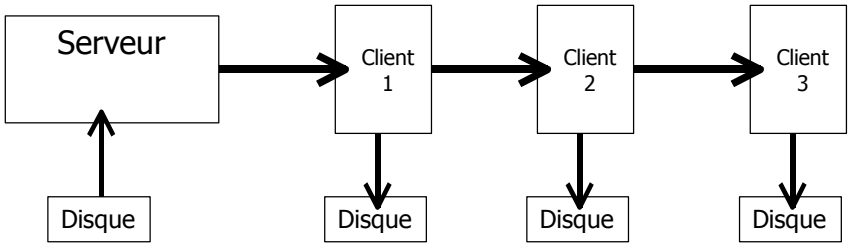

Figure 5 : chaîne d'installation du système d'exploitation

L'application résultante se nomme Ka et est disponible en téléchargement sous licence GPL. L'installation de machines avec Ka se passe de la manière suivante : on utilise PXE/BpBatch pour démarrer un mini système Linux. Ce mini-système monte son système de fichiers racine par NFS à travers le réseau. Ensuite il crée et formate les partitions qui vont être utilisées pour installer le système d'exploitation, puis lance le client Ka. Le client Ka va se connecter à un serveur Ka situé sur la machine à cloner. Et lorsque toutes les machines à installer se sont connectées au serveur Ka, celui-ci coordonne la création d'une chaîne de connections TCP. Une fois cette chaîne créée, elle est utilisée pour diffuser vers toutes les machines les données constituant le système à installer. Les données sont alors lues depuis le disque dur de la machine à cloner, envoyées vers les machines à installer et écrites sur le disque dur de celles ci en continu.
Une fois toutes les données écrites sur le disque dur, la machine peut redémarrer et utiliser le système qu'elle vient d'installer. Afin de ne pas recommencer la procédure d'installation au début lors de ce redémarrage, on maintient à jour sur un serveur l'état d'avancement de

l'installation pour chacune des machines. Ce fichier est consulté par BpBatch lors du démarrage afin de choisir l'action à effectuer (démarrer l'installation ou démarrer le système qu'on a installé ?). Ce fichier est mis à jour au fur et à mesure des étapes de l'installation. Dans notre cas, ce fichier peut contenir trois états différents : système encore à installer, système installé mais secteur de « boot » pas encore écrit, système prêt. Chacun de ces états correspond à une action : installation de la machine, écriture du secteur de « boot », démarrage de la machine pour la production.
Voici pour résumer un aperçu des échanges faits à travers le réseau lors de la première étape de l'installation.

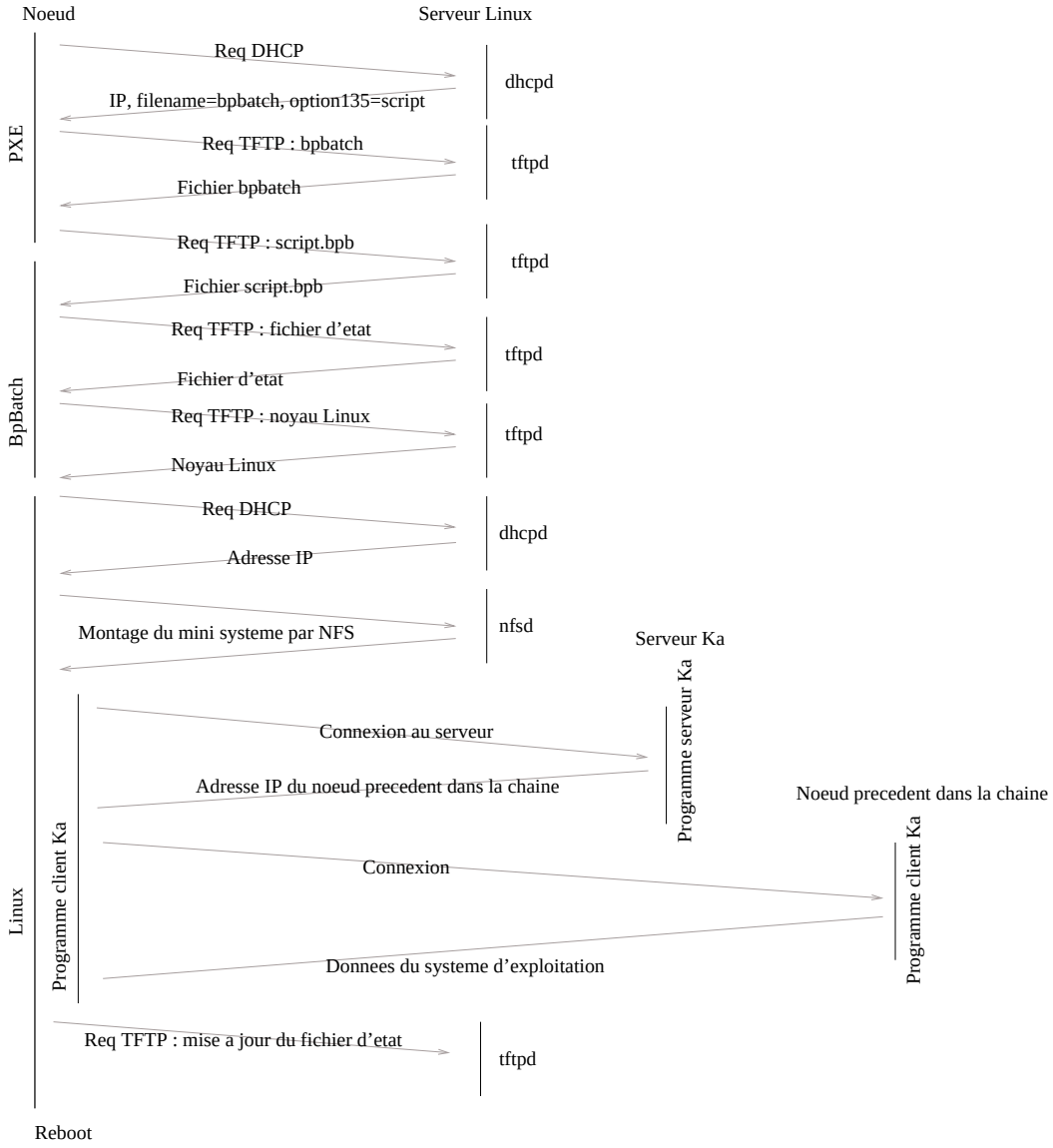

Figure 6 : processus de démarrage par le réseau

LE PROBLEME DU PASSAGE A L'ECHELLE

En terme de performances, le problème des grappes de PC et des environnements de calcul parallèle est de fournir une puissance de calcul pratiquement proportionnelle au nombre de machines. En terme d'outils d'exploitation, il s'agit cette fois de maintenir un temps de réponse constant ou au moins raisonnable quel que soit le nombre de machines.
Le laboratoire ID a développé des réponses en s'appuyant sur des logiciels standards, MPI et Posix dans le cas de la programmation parallèle, *NFS*, *rsh*, *rcp* ou la diffusion multicast dans le cas des outils d'exploitation.

Diffusion de données :

La copie d'un fichier sur toutes les machines du cluster (besoin commun aux utilisateurs et administrateurs) peut être réalisée sous forme de diffusion multicast ou de diffusion unicast en arbre. Comme pour le déploiement de système d'exploitation, la copie d'un fichiers de grande taille utilisera une chaîne optimisée de copies séquentielles.

Lancement de commandes :

L'objectif d'un lanceur de commandes est non seulement de démarrer rapidement les processus qui interviennent dans une application ou une commande parallèle mais aussi de véhiculer proprement les signaux entre processus et enfin gérer efficacement le flux des entrées/sorties.
La solution développée au sein du laboratoire consiste d'une part à réutiliser les démons *rshd* standards et à paralléliser le client *rsh*. Cette parallélisation est d'ailleurs double, il s'agit de paralléliser les trois phases classiques du *rsh* (accès à la machine distante, authentification, mise en place de la connexion) d'une part, et de réaliser des appels récursifs à la commande *rsh* de manière à recouvrir les machines concernées par un arbre.
Cette phase de lancement crée ainsi un arbre de connexions TCP qui peut être utilisé pour véhiculer entrée/sorties et signaux. L'utilisation de cet arbre dans les environnements de monitoring et les librairies de passages de messages s'avère très performant.

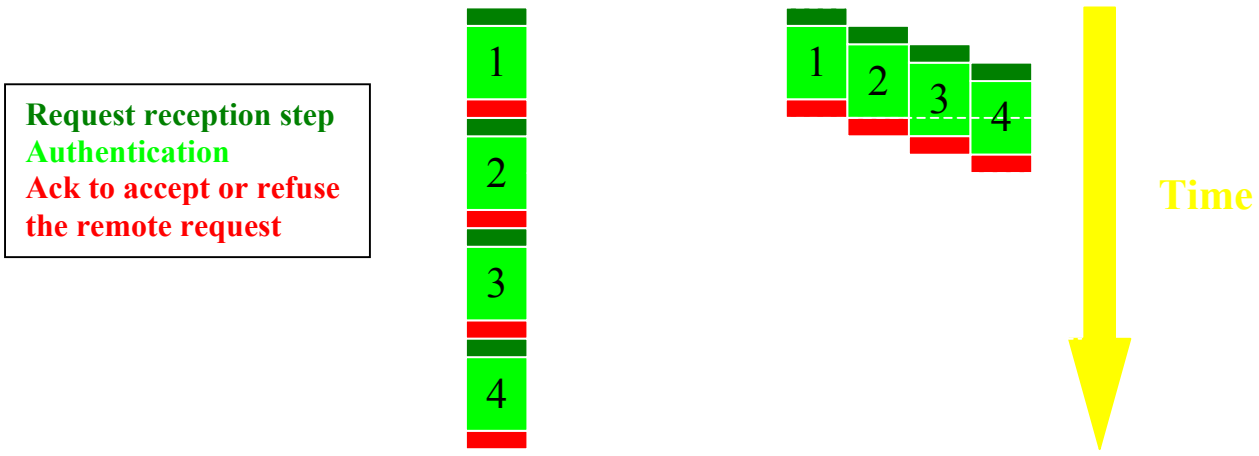

Figure 7: le temps d'authentification est pipeliné

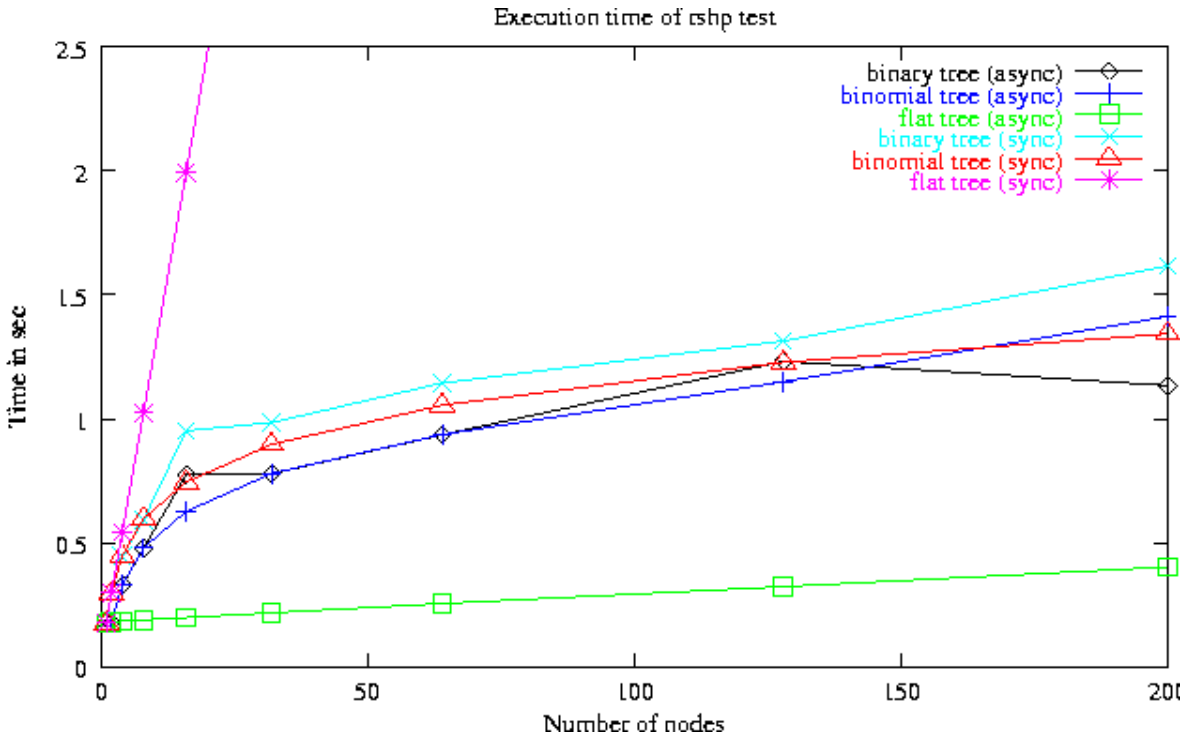

Figure 8 : Temps de lancement en fonction de la structure de diffusion

Système de fichiers

Les performances de NFS peuvent se dégrader à partir d'une vingtaine de clients. Une recommandation aux utilisateurs est de déployer les fichiers sur les disques locaux des machines de la grappe à l'aide de la commande de copie performante décrite précédemment. Une alternative plus confortable à la distribution de fichiers est la mise en place de systèmes de fichiers qui soutiennent le passage à l'échelle. Des évaluations de système existants comme CODA ou PVFS [8] se sont révélées décevantes dans notre contexte de travail. Une autre approche est donc en étude au laboratoire ID. Le but de ce travail est de proposer un système distribué de fichier pour clusters, c'est à dire en environnement sécurisé, homogène, avec un réseau reliant les diverses machines équipées de disques, tout en minimisant les coûts d'installation et de mise en œuvre. Le problème actuellement est de disposer d'un espace commun de stockage pour tous les nœuds en utilisant les nœuds eux même afin de repartir la charge d'E/S. A moins d'avoir un matériel dédié onéreux (systèmes SAN), il y a peu de solutions à part d'essayer de répartir la charge en créant plusieurs sous serveurs. Plutôt que de concevoir "from scratch" un système de fichiers distribués (PVFS [8]), l'idée est de s'intégrer dans l'existant. Dans beaucoup de systèmes c'est encore NFS qui est présent, malgré ses défauts (pas de cohérence stricte, pas de contrôle de la congestion [UDP]). Evidemment cela simplifie le développement puisque la partie client existe déjà. Il suffit de modifier le serveur de manière transparente pour le client : le serveur va devenir un serveur de méta-données (taille, numéro du fichier, permissions Unix, etc.); Les E/S lourdes (lecture, écriture) quant à elles seront assurées par des processus situés sur d'autres machines. L'avantage de cette approche est que le "coût" d'administration et d'installation est faible puisque l'installation se limite à lancer les démons d'E/S, lancer le serveur, et monter la partition de type *nfs*. (pas de recompilation du noyau, pas de repartitionnement). Cette solution est en court de test sur le I-Cluster [1].

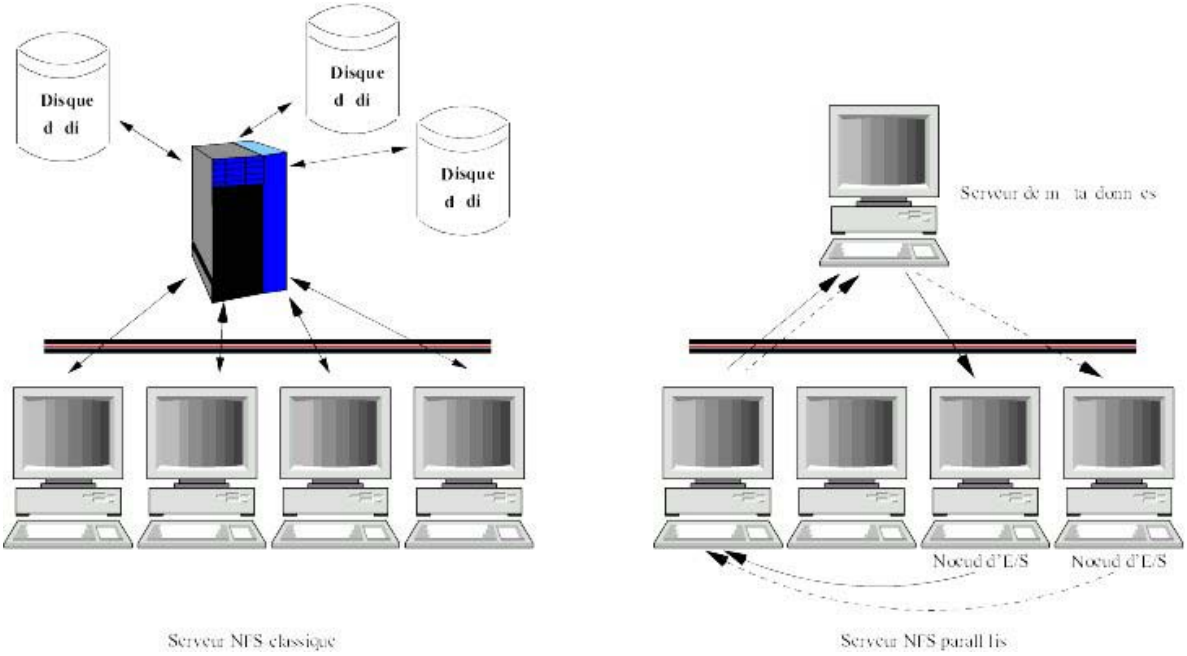

Figure 9 : l'architecture NFS parallèle

MONITORING

Des outils de supervision da la grappe sont en développement. Il s'agit de visualiser des informations systèmes (charge, mémoire disponible, trafic réseau…) sur tous les nœuds. Pour cela, on doit combiner un outil de collecte locale d'information, un outil pour véhiculer les informations sur le réseau et un outil de visualisation. Un grand nombre d'outils existent pour la collecte locale d'informations systèmes. Nous utilisons Performance Co-Pilot de SGI.
Pour véhiculer les informations sur le réseau, on peut utiliser le flot d'entrée/sortie du lanceur de commandes décrit plus haut.
La visualisation est le point le moins mature. Il s'agit d'accéder à l'information pertinente dans de gros volumes d'information et de pouvoir traiter des informations en provenance de sources variées (systèmes, applicatives, réseau). Pajé, est en outil en cours de développement au laboratoire ID, qui vise à répondre à ces deux problèmes par l'utilisation de filtres interactifs (pour la visualisation des informations importantes) et par l'utilisation d'un format de données générique (le langage sépare la description des données et les données elles-mêmes) qui permet de visualiser des données de toutes origines [12].

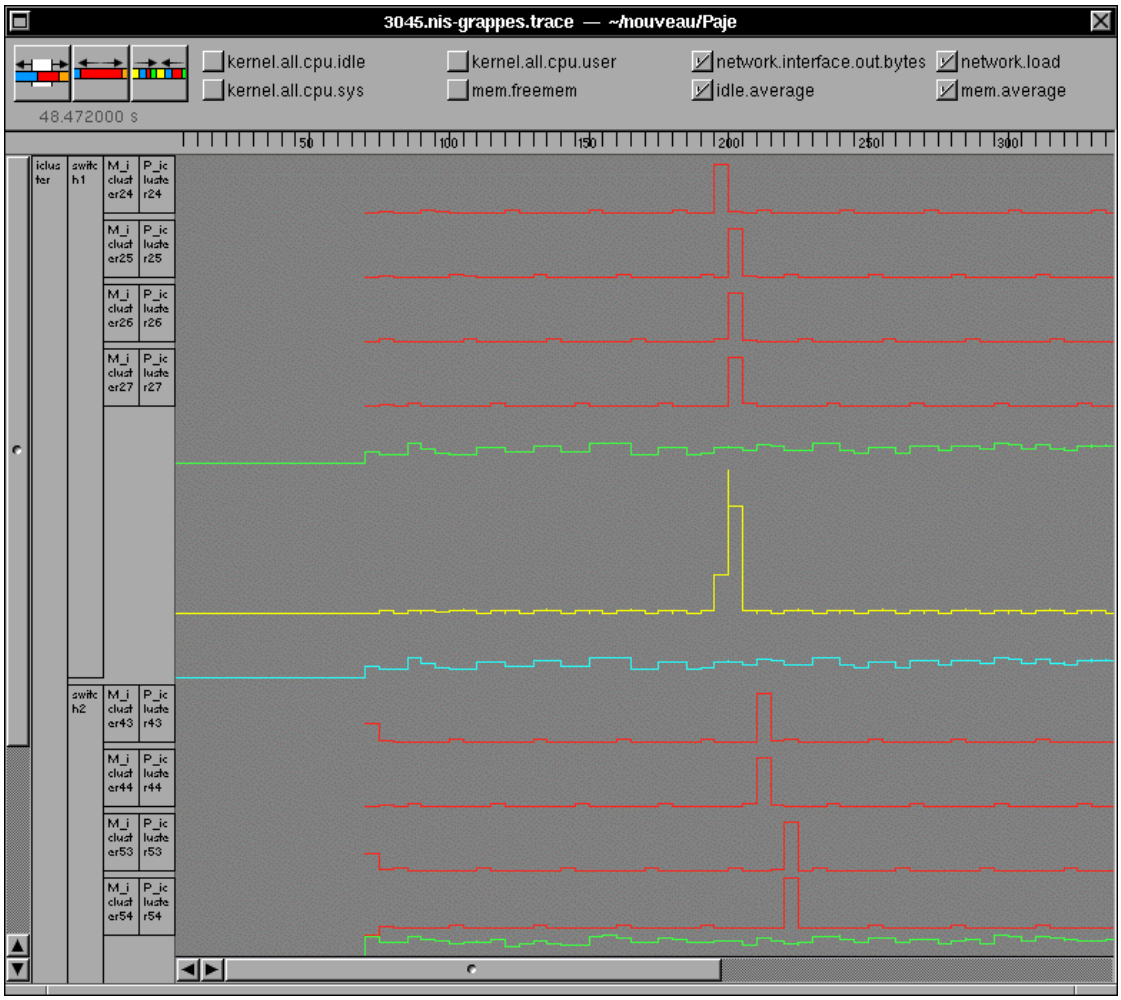

Figure 10 : une capture d'écran de Pajé

CONCLUSION/PERSPECTIVES

Nous avons présenté la mise en exploitation d'une grappe de PC, en particulier le problème du déploiement automatique et rapide des nœuds de calculs. Ce travail est une brique de base à la réalisation d'une distribution Linux pour grappes de PC, projet RNTL commun au laboratoire ID et aux sociétés Bull et Mandrakesoft. En plus du passage à l'échelle discuté dans cet article, les objectifs du projets sont la compatibilité avec des matériels variés, la mise à disposition d'interface graphique d'utilisation et d'administration et enfin l'amélioration des environnements de calcul parallèle existant. Mais les outils d'exploitation développés pour la grappe de PC peuvent être aussi utilisés dans d'autres contextes.
Le développement du metacomputing (calcul sur Internet de type ***SETI@home*** [5] ou bien calcul sur grilles de grappes) demande que les procédures d'exploitation de grappes soient ouvertes à une utilisation extérieure et hétérogènes des ressources informatiques. Le développement d'un portail applicatif (interface web d'utilisation de la grappe sans compte Unix) et d'un allocateur/scheduleur de ressources pour grilles de grappes sont parmi les sujets sur lesquels se sont orientés les développements du laboratoire. Enfin, l'outil de déploiement automatique de système a été étendu à la prise en compte du système Windows2000 puis à l'utilisation de machines à double amorçage Windows2000/Linux. Nous avons aussi développé le logiciel permettant à un utilisateur de redémarrer sa machine en mode grappe sous Linux pour une durée de son choix. Ceci devrait permettre de déployer dans les Intranet d'entreprises des grappes de PC virtuelles conciliant une utilisation bureautique et une utilisation calcul.

Références :

[1] Pierre Lombard. Serveur NFS parallélisé pour une grappe de PC, communication interne, Août 2001.
[2] `Alinka` : http://www.alinka.com/
[3] `Scyld` : http://www.scyld.com/
[4] `Oscar`, Open Source Cluster Application Resources : http://www.csm.ornl.gov/oscar/
[5] A new major SETI project based on Project Serendip data and 100,000 personal computers. W. T. Sullivan, III, D. Werthimer, S. Bowyer, J.Cobb, D. Gedye, D. Anderson. Published in: "Astronomical and Biochemical Origins and the Search for Life in the Universe", Proc. of the Fifth Intl. Conf. on Bioastronomy. 1997
[6] `BpBatch : http://www.bpbatch.org/`
[7] P. H. Carns, W. B. Ligon III, R. B. Ross, and R. Thakur, ``PVFS: A Parallel File System For Linux Clusters" (postscript, html), Proceedings of the 4th Annual Linux Showcase and Conference, Atlanta, GA, October 2000, pp. 317-327
[8] I-Cluster: Reaching TOP500 performance using mainstream hardware, B. Richard, P. Augerat, N. Maillard, S. Derr, S. Martin, C.Robert
[9] C. Martin, O. Richard. Parallel launcher for clusters of PC, submitted to World scientific
[10] SGI Silicon Graphics Inc. Linux advanced cluster environment. http://www.sgi.com/federal/solutions/linuxclustering.html, http://oss.sgi.com/projects/ace/.
[11] C3, Cluster Command and Control, http://www.epm.ornl.gov/torc/C3/
[12] C. Guilloud, J. Chassin de Kergommeaux, P. Augerat, B. Stein , Outil visuel d'administration système pour grappe de processeurs de grande taille, RENPAR 2001, 24, 27 avril 2001.
[13] `PBS`, www.OpenPBS.org. The Portable Batch System Software (PBS v2.2p13), Veridian, PBS Products Dept., Mountain View, CA, March 2000.
[14] HPL http://www.netlib.org/benchmark/hpl/, http://www.hpl.hp.com/techreports/2001/HPL-2001-206.html