

Les grilles informatiques - état de l'art et déploiement

Marcel Soberman
CNRS / STIC
marcel.soberman@cnrs-dir.fr

Résumé

Le concept de grilles informatiques a fait ses preuves, des grilles de production sont maintenant opérationnelles en entreprises et entre organismes. Au delà d'une solution pour les calculs intensifs, l'approche grille permet la constitution d'organisations virtuelles (VO) regroupant des communautés d'utilisateurs partageant des ressources globalisées de traitement, de stockage, des instruments et des services largement distribués, quelles qu'en soient les organisations propriétaires.

Le déploiement de grilles a donc un impact fort sur le plan organisationnel, les réseaux, la sécurité. L'article présente:

- les principales notions
- les architectures de grilles, les principales voies technologiques, leurs évolutions
- les problématiques encore insuffisamment traitées
- les impacts sur l'administration des réseaux et des services
- les applications concernées
- les principaux acteurs en France.

Mots clefs

Grille informatique, Traitement distribué, Grid computing, Calcul intensif, Virtualisation

1 Objectif de l'article

Sous des dénominations diverses: grid computing, data grid, informatique 'on demand', utility computing, on trouve souvent l'usage de grilles informatiques.

Ce concept était à la base essentiellement technique, avec pour objectif principal la mise à disposition de ressources de traitement informatique très importantes et multi organismes avec un accès aussi transparent que possible. Le concept a fait ses preuves autour d'infrastructures prototypes; des grilles de production sont maintenant opérationnelles en entreprises et entre organismes. Le déploiement d'infrastructures de ce type en production a des impacts forts sur le plan des réseaux, de la sécurité, de l'administration au sein de chaque organisme, et sur le plan organisationnel puisqu'un l'un des objectifs visés est celui du partage de ressources entre entreprises ou institutions.

Cet article présente l'état de l'art, les déploiements remarquables actuellement en cours avec leur impact en

2005. Cet article, forcément court, reprend et actualise des parties du livre « Les grilles informatiques » publié en 2003 [20] , en sélectionnant les points sur lesquels il m'est apparu important d'apporter une connaissance de base à des administrateurs systèmes et réseaux.

2 Introduction aux grilles

En 1998, Ian Foster & Karl Kesselman ont posé le paradigme de la grille informatique dans leur ouvrage: « *The Grid: Blueprint for a New Computing Infrastructure* » [17] . L'accès aux services informatiques devrait être comparable en coût et facilité au branchement d'appareils sur le réseau électrique. Cet accès devrait être aisé et transparent quels que soient la puissance nécessaire, la complexité des équipements matériels et logiciels mis en œuvre, les fournisseurs du service.

Derrière cette image se cachent des concepts puissants. Le rapprochement avec d'autres concepts récents comme les services Web, l'informatique à la demande, le Web sémantique conduiront à une évolution importante dans l'usage des ressources informatiques.

En effet, au delà des besoins en capacité de traitement, l'approche grille permet la co-allocation et le partage dynamique de toutes les ressources de systèmes hétérogènes indépendamment des unités organisationnelles (départements d'entreprises, institutions et entreprises). Des organisations virtuelles (VO) permettent maintenant de définir en tant que de besoin des communautés d'utilisateurs partageant suivant des modalités communes d'utilisation, des sous-ensembles de ces ressources globalisées de traitement, de stockage, des logiciels, des instruments et des services largement distribués, quelles que soient les organisations propriétaires.

3 Les technologies

3.1 La technologie « Massive Grid »

Le terme GRID (Globalisation des Ressources Informatiques et des Données) est souvent donné à la technologie d'intergiciel support des grilles dont les nœuds sont constitués par des grappes de serveurs ou des calculateurs de grande puissance localisés dans des centres de calcul différents, ces nœuds étant reliés sur des réseaux haut débit, en VPN pour des raisons de sécurité. Un premier usage de cette technologie, distribuée depuis longtemps par SUN (Sun Grid Engine) ou Hewlett Packard (HPC utilities) a été l'optimisation des serveurs au sein

d'une même entreprise. Elle permet de constituer des grilles de forte puissance avec des niveaux de sécurité acceptables dans bien des cas grâce au verrouillage des accès, mais aussi parce que les centres de calcul sont censés disposer d'administrateurs systèmes et réseaux et d'une politique de sécurité. Des exemples de grilles de ce type, opérationnelles à très large échelle, sont constitués par les projets:

- DEISA, constitué autour de neuf sites dotés de super-calculateurs (Figure 1):



Figure 1 - Les sites du projet DEISA

avec une architecture particulière constituée autour du système de gestion de fichiers unique GFS (Global File System) délivré par IBM (Figure 2):

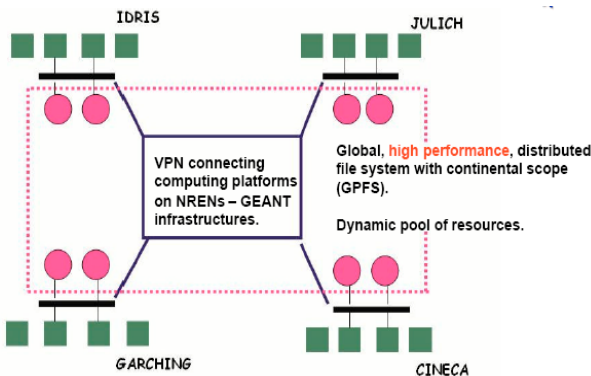


Figure 2 - Architecture DEISA Phase 1 (4 premiers sites, 4000 processeurs, 24 TFlops)

- EGEE (Figure 3) interconnectant plus de 180 centres de calcul scientifiques européens et mondiaux, initialement à partir d'une large extension de Globus (<http://www.globus.org/toolkit/>) dans le cadre du projet européen DATAGRID (<http://eu-datagrid.web.cern.ch/eu-datagrid/>), porté dans

l'intergiciel LCG (<http://lcg.web.cern.ch/LCG/>) qui constitue le cœur d'un intergiciel en gLite [4] orienté services web, et dont les premières versions sont disponibles:

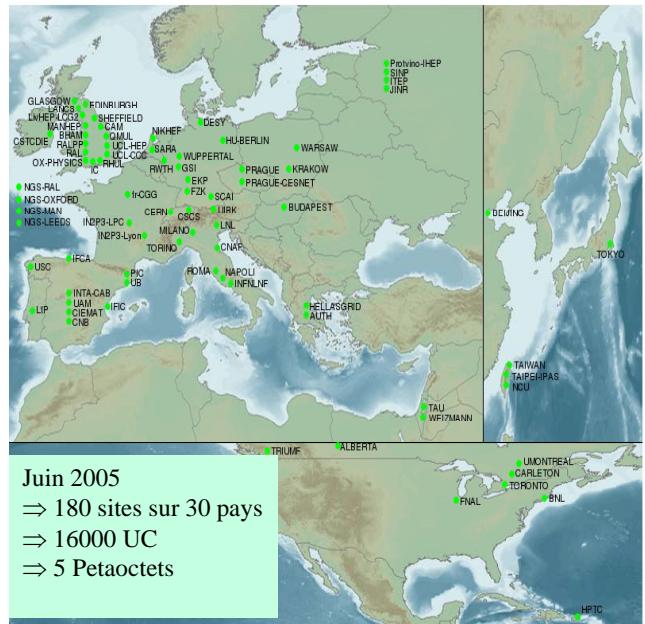


Figure 3 - EGEE

3.2 La technologie « Desktop Grid » ou « Internet Computing »

Une autre technologie de grille a vu le jour à partir de l'observation des outils P2P (peer to peer) très utilisés par nos gamins pour se transférer de la musique ou des films via Internet. L'application travaille en client-serveur suivant le schéma de la Figure 4, sous le pilotage d'un serveur de grille:

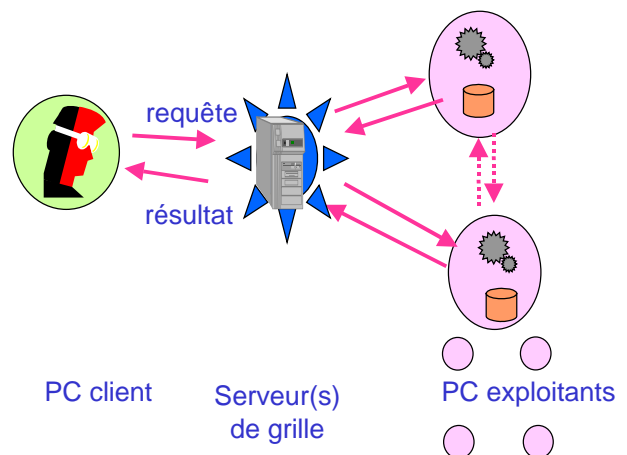


Figure 4 - Principe de grille pair à pair en Desktop Computing

Dans cette technologie pair à pair entre des PC fournisseurs et un PC client (qui peut lui-même être

fournisseur) [2] l'exécution des codes est généralement automatisée, notamment sur déclenchement par l'écran de veille lorsque la machine n'est pas utilisée, la recherche des machines disponibles s'effectuant suivant les systèmes en mode push ou pull. Cette technologie, le Desktop Grid (dénommée aussi: Internet Computing ou Global Computing) est à même de piloter des centaines de milliers, voire des millions, de PC représentant des dizaines de Teraflops via Internet comme on a pu le constater dans des projets majeurs comme Seti@home pour la recherche de l'intelligence dans l'univers et le Decryphon pour la génomique en France.

Cette technologie souffre de trois grands écueils:

- les applications sont limitées par la taille des machines utilisées et le débit des réseaux en transfert de données
- la sécurité d'utilisation est très faible ; c'est pourquoi les entreprises réservent cette technologie à un usage interne sur leur Intranet
- la fiabilité du système global: à tout moment, une ou plusieurs machines sont en panne, éteintes, ou simplement reprises en usage normal par leur utilisateur principal alors qu'un travail était déclenché dessus par la grille! La reprise de ce travail et sa relance sans risque de doublon constituent un des critères de choix des intergiciels du secteur.

Par contre, cette technologie est facile à déployer, et à un faible coût, si on lève les problèmes de pare feux. Des intergiciels Open Source comme Xtremweb (<http://www.xtremweb.net>) et des produits commerciaux sont largement disponibles. On la trouve donc maintenant fréquemment dans certains secteurs d'activité, comme la R&D pharmaceutique.

3.3 Autres technologies

Les deux technologies citées sont les plus répandues. D'autres bases technologiques méritent d'être étudiées, comme ComputeMode de la société Icatiss (<http://www.icatiss.com>) qui permet de basculer la nuit, le week-end ou en tant que de besoin des machines sous un même OS afin d'offrir une puissance complémentaire à une grappe de serveurs.

4 Architectures de grilles

4.1 Les intergiciels (middlewares) de grille

La Figure 5 présente une vision simplifiée d'une architecture générique, c'est à dire quelle que soit la technologie utilisée, de grilles:

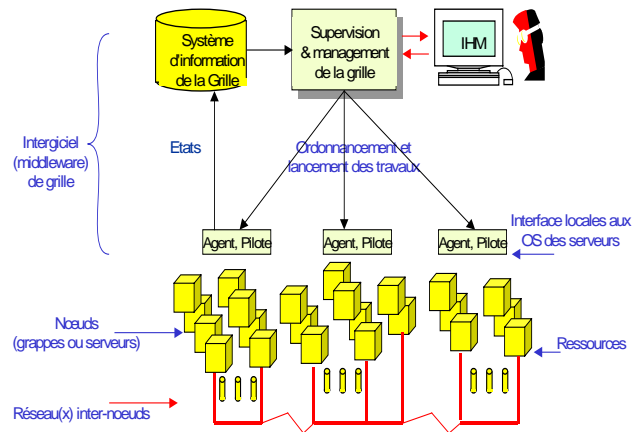


Figure 5 - Architecture simplifiée de grille

La grille est physiquement constituée de **nœuds**, qui sont des processeurs avec leurs disques, l'ensemble étant interconnecté via un **réseau**, dont la qualité peut être primordiale suivant la technologie de grilles retenue. Suivant la voie technologique retenue, ces nœuds sont de serveurs plus ou moins puissants, voire des PC, ou des grappes de serveurs (clusters).

Un logiciel d'interface et de pilotage est installé sur chaque nœud. Il assure le lien entre l'activité locale du nœud, supervisée par les systèmes d'exploitation de chaque serveur, et un outil de supervision et de management global de la grille logiquement unique, mais généralement distribué sur plusieurs machines pour une meilleure fiabilité. L'ensemble des logiciels assurant la gestion de la grille est dénommé **l'intergiciel** (middleware) de la grille. Celui-ci gère toutes les **ressources** de la grille, est informé en permanence de leur état: unités de traitement et de stockage constitutives des nœuds, branches de réseau, bibliothèques de programmes... toute entité qui peut être définie et administrée par le gestionnaire de ressources. Il assure aux utilisateurs et aux applications les services de:

- **réservation et allocation** des ressources nécessaires
- **ordonnancement et lancement des travaux**
- **suivi** de l'activité
- **administration** du système.

Des utilitaires adaptés permettent d'assurer des fonctions optimisées pour les grilles. L'exemple le plus connu est GridFTP pour la transmission de fichiers en tenant compte de leur localisation multi-nœuds, avec des possibilités multicanaux si le réseau le permet.

4.2 Evolution des intergiciels

Ce que nous venons d'exposer correspond à une première génération permettant de soumettre des travaux en traitement différé sur une grille, souvent après avoir placé *a la mano* les programmes et données sur les nœuds disponibles. Cette architecture est toujours utilisée, ne serait ce que pour bâtir des grilles dites légères, cet adjectif

soulignant leur facilité d'implantation et d'administration en liaison avec des gestionnaires locaux de batch sur grappes de serveurs comme OpenPBS, MAUI ou autre.

La deuxième génération apporte un outil essentiel d'automatisation dans l'usage de grilles, le **gestionnaire de ressources** (resources broker) schématisé Figure 1:

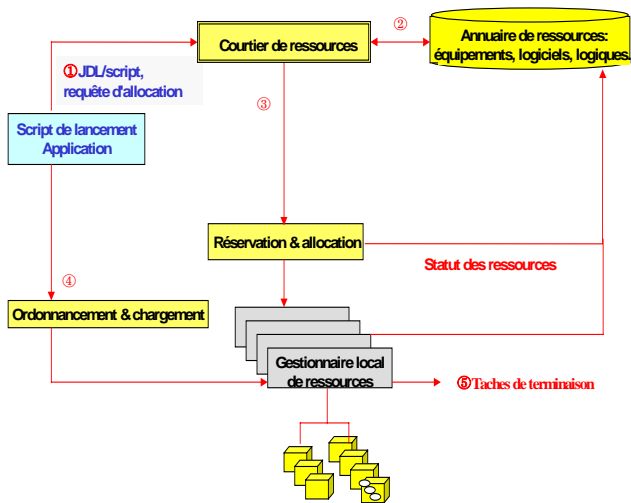


Figure 6 - Le courtier de ressources (resource broker)

Celui-ci dispose de l'état de l'ensemble des ressources offertes par la grille, ce qui lui permet d'allouer celles demandées par les scripts de lancement des travaux et disponibles, voire de les réserver préalablement, avant ordonnancement puis exécution.

La 3^{ème} génération se met en place. Les points majeurs en sont:

- une meilleure sécurité d'accès par l'utilisation systématique de **certificats électroniques**
- une gestion des fichiers adaptée à la prise en charge de données massives sur grille, avec découpage et regroupement en **entités logiques** pouvant être **répliquées** en tant que de besoin pour des traitements parallélisés avec une localisation proche
- la généralisation de la notion d'**organisation virtuelle** permettant de définir des sous ensembles de ressources et leurs modalités d'usage par des communautés d'utilisateurs, et ceci indépendamment des notions de comptes et d'appartenance juridique
- l'introduction de **modèles économiques** [5], [6], la prochaine étape allant vers l'**accès sémantique** aux services: trouver le service qui assurera un traitement sur tel objet en moins de n minutes
- le passage d'un mode accès à des ressources d'une grille vers la fourniture de **services de grilles** (web et grid services)
- l'interopérabilité via une normalisation des **services web appliqués aux grilles**: OGSA [1] et WS-RF (Web Services Resource Framework) spécifié par le consortium normatif Oasis

- des services **réseaux**: protocoles plus performants pour le multicast, les transferts massifs de données et pour les messages type MPI (Message Passing Interface), allocation de bande passante à la demande
- des approches basées sur les **SLA** (Service Level Agreement) au niveau réseau et au niveau grille
- une meilleure prise en compte des **besoins industriels**: administration simplifiée, sécurité, facturation des services et business models, interopérabilité des grilles, fiabilisation des services, connexion à la demande à la grille, applications interactives et transactionnelles, etc...

4.3 La soumission des travaux

La soumission d'un travail simple sur une grille via un script en JDL (Job Description Language de DATAGRID ou EGEE) est présentée ci-dessous:

Soumission = edg-job-submit --vo <nom de VO> <nom du fichier.de JDL>

Exemple de fichier script en JDL sur EGEE:

```
Executable           = "/bin/gridTest";
StdError             = "stderr.log";
StdOutput            = "stdout.txt";
InputSandbox         = {" /home/joda/test/gridTest" };
OutputSandbox       = {" stderr.log", "stdout.txt" };
InputData            = "lfn:testbed0-00019";
DataAccessProtocol   = "gridftp";
Requirements         = other.Architecture=="INTEL"
&& other.OpSys=="LINUX" && other.FreeCpus >=4;
```

On remarquera la sélection des ressources utilisables dans la ligne Requirements.

Comme nous l'avons dit, cet exemple présente un cas simple, et les utilisateurs ont de plus en plus besoin de définir des flots de travaux conditionnés par l'exécution de tâches préalables, grâce à des outils de workflow.

De plus en plus, la complexité est cependant masquée aux utilisateurs par des portails permettant de décrire graphiquement les travaux, déclenchant des scripts pré paramétrés et permettant de visualiser le suivi de l'exécution, offrant ainsi l'accès transparent à une ou plusieurs grilles de machines hétérogènes. Ils pourront aisément s'adapter pour offrir l'interface à des services Web. La Figure 7 montre un exemple de soumission de plusieurs travaux via le portail GENIUS développé par l'INFN (Istituto Nazionale di Fisica Nucleare):

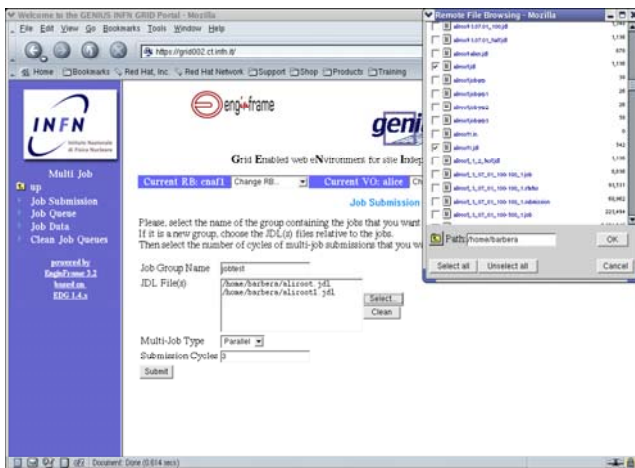


Figure 7 - Portail GENIUS développé par l'INFN

Il est à noter que ce portail dispose aussi d'une version pour PDA!

La Figure 8 montre les composants traversés lors de la soumission d'un travail, ici sur DATAGRID, avec l'activation des fonctions de:

- autorisation et authentification
- courtage des ressources à partir des informations sur leur disponibilité
- d'information sur l'état des travaux (logging & bookkeeping)
- soumission via Globus vers les computing elements (ensemble de processeurs, en général une grappe de serveurs) et les storage elements (serveurs de stockage)
- gestion des fichiers via un catalogue des différentes répliquations de celui-ci (la localisation des données à proximité des programmes est ici manuelle).

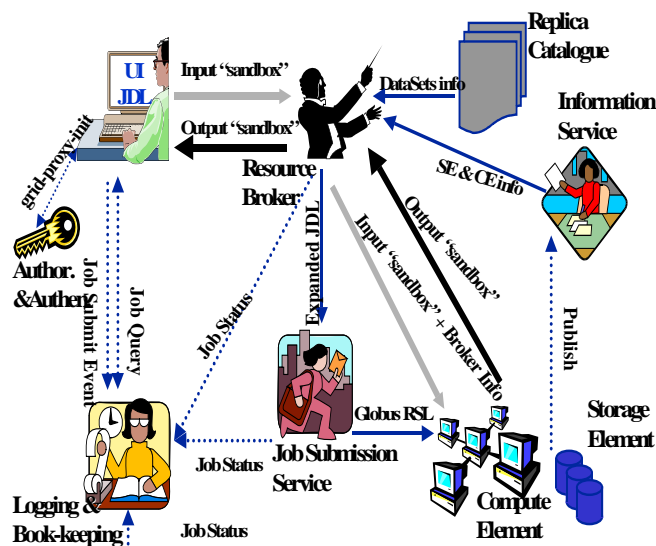


Figure 8 - La soumission de travaux sur DATAGRID

4.4 Sécurité et confidentialité

La sécurité constitue un des points majeurs pour l'implantation des grilles, et ceci d'autant plus qu'un des objectifs est de partager des ressources informatiques. C'est typiquement le cas dans l'usage de PC bureautiques de divers départements d'une entreprise, ou de l'informatique à la demande inter organismes: les logiciels et les données peuvent se trouver ainsi sur des nœuds accessibles à des tiers.

Un premier niveau de sécurité est apportée dans l'accès à la grille elle-même, nécessairement de type SSO (single sign on) pour ne pas avoir à saisir tous les identifiants et mots de passe d'accès chaque nœud pour l'exécution du moindre travail. La généralisation des certificats électroniques permet une sécurisation généralement satisfaisante dans les deux sens, c'est à dire aussi bien pour protéger l'accès aux ressources que pour garantir aux utilisateurs que les serveurs accédés appartiennent bien à la grille¹. Ceci n'est vrai que si des politiques de sécurité adéquates existent aux niveaux locaux et qu'elles sont cohérentes au niveau global de la grille.

Par contre, les données et les logiciels sont souvent moins bien protégés. Leur sécurisation passe par:

- la mise en œuvre des ACL (Access Control List) au niveau des fichiers logiques et physiques
- le cryptage des fichiers demeurant sur les serveurs, ce qui est coûteux à chaque utilisation et pas totalement sûr
- pour les codes logiciels, l'utilisation de techniques de « sandbox » pour éviter les attaques via les applications, et d'obfuscation [12] ; des dispositifs sont prévus à cet effet en Java et dans .Net au niveau JVM (Java Virtual Machine) et CLR (Common Language Runtime chez Windows).

Lorsqu'on utilise du desktop computing sur des PC non contrôlés, la pratique a montré la nécessité de bien contrôler les exécutions suivies par la comparaison des résultats entre eux ou en comparaison d'un résultat canonique, ou d'un contrôle statistique des résultats afin de détecter les modifications du code par les possesseurs des PC. On trouvera des approches de certification d'applications dans cet environnement notamment dans [13].

Le cryptage n'est guère possible pour les bases de données [14] : on ne peut définir les colonnes de tables et les indexer sans donner de l'information sur le cryptage. En revanche, on peut observer que l'accès aux données des bases relève de deux types: le traitement de masse, par

¹ Voir notamment le logiciel CNRS-IGC développé par l'UREC (<http://www.urec.cnrs.fr/igc/CNRS-IGC/>) et le site Eu Grid PMA (<http://www.eugridpma.org/>) pour la mise en place des autorités de certification

exemple pour des statistiques ou des transferts, et l'accès sélectif. Des solutions sont étudiées aujourd'hui pour préserver la confidentialité, particulièrement dans le domaine biomédical [15] [12] :

- liaison avec des serveurs de confiance permettant d'établir la sécurité des échanges entre une application et l'acquisition des données
- anonymisation des données avec retro lien permettant la traçabilité sur demande explicite et l'accès autorisé aux données de base
- anonymisation des transactions.

La généralisation d'hyperviseurs -superviseurs de machines virtuelles logiques- permet d'envisager des solutions prometteuses sur le plan de la sécurité, tant par la séparation d'accès à chacune d'entre elles que par les possibilités d'introduire des processeurs de codes particuliers avec une clé par propriétaire de la machine virtuelle.

4.5 Le déploiement

4.5.1 Déploiement de composants de l'intergiciel

Le déploiement d'un intergiciel par l'installation sur chaque machine des composants nécessaires n'est possible que sur des grilles de petite taille.

L'auto installation sur des Desktop Grids est effectuée couramment. Il est vrai qu'il s'agit de composants aux fonctionnalités limités, et cela relève fréquemment de scripts tels que ceux servant aux mises à jour par les administrateurs de parc bureautique.

Des outils de déploiement ont été conçus pour automatiser le processus de déploiement à large échelle, comme Kadeploy (<http://ka-tools.imag.fr/>) pour les machines de fermes de PCs. Un redémarrage à distance est même possible sous certaines conditions, et est en service par exemple sur la plate-forme d'expérimentation Grid5000 (<http://www.grid5000.org/>).

4.5.2 Déploiement des applications et des logiciels applicatifs

Le déploiement des applications n'est pas trivial lorsqu'on considère tous les cas possibles. Quels sont les sous-ensembles persistants sur tel ou tel nœud, que doit-on ôter en fin d'un travail et réinstaller lors du prochain lancement (rappelons que les programmes devront peut être s'exécuter sur d'autres machines que la fois précédente suivant les disponibilités, et que la localisation à proximité des données est souhaitable)?

D'autres problèmes sont posés par les licences associées aux logiciels, au delà du problème de leurs prix lorsqu'ils sont dépendants du nombre de processeurs utilisés. Ce point n'est pas que commercial, comme le croient certains. Payante ou gratuite, une licence est avant tout un contrat

décrivant des règles d'utilisation: par exemple, seuls des utilisateurs académiques sont autorisés à utiliser ce logiciel et seulement sur 10 exécutions simultanées sur des sites appartenant à tel organisme. Ce qui est déjà difficile à appliquer devient un casse-tête si l'intergiciel de grille ne comprend pas les outils appropriés, et la mise en place de serveurs de licences tel FLEXLM se heurte de plus à des problèmes classiques de pare feux.

4.6 Impact sur les réseaux

Un gros effort a été accompli sur les performances d'interconnexion locale de grappes de serveurs avec des technologies comme Myrinet ou Infiniband. Les grandes grilles doivent traverser des WAN, avec les problèmes de bande passante et surtout de latence liés, ce qui induit des difficultés dans la prévision des délais d'exécution de travaux et surtout dans l'architecture des applications. Les évolutions envisagées passent par:

- un suivi au niveau de la grille de l'ensemble en interaction avec les opérateurs (NREN dans le cas de la recherche académique), la création de services de bout en bout encadrés par des SLA (Service Level Agreement) globaux reposant eux-mêmes sur des SLA réseaux:
 - réservation et allocation à la demande de bande passante par les utilisateurs, donc les applications
 - QoS multi domaines avec des points d'entrée permettant aux applications d'activer des services réseaux en tant que services de grille
- la connaissance de la disponibilité réelle de chaque branche du réseau pour une allocation optimale des ressources
- la mise en place de réseaux actifs avec des serveurs capables de filtrer les données afin de limiter les volumes de transfert inter nœuds, assurant le pilotage des QoS temporairement nécessaires, et l'utilisation de dépôts de données au sein des réseaux.

4.7 Administration, supervision et métrologie

4.7.1 Administration et supervision

L'administration d'une grille demande un effort particulier. Au delà du paramétrage local des serveurs et des réseaux utilisés, on trouve celui de l'intergiciel de grilles et des outils associés, celui des pare feux, et l'administration commune de sites, d'utilisateurs appartenant parfois à de multiples organismes avec leur règles spécifiques. Des outils de supervision comme MapCenter permettent le suivi de l'état des sites, du réseau et des serveurs.

Les grilles de très grandes tailles obligent à une structuration en niveaux, un peu comme ce qu'on trouve pour les réseaux avec GEANT et les NREN. L'exemple de

la structuration de l'administration de EGEE est donné Figure 9, avec:

- un pôle principal, l'OCC (Operations Coordination Centre), localisé au CERN en liaison avec des centres de service à tour de rôle
- des pôles régionaux (9 aujourd'hui): les ROC (Regional Operation Centres) qui coordonnent le déploiement et la production
- le GGUS (User Support Centre), point de contact unique pour le support utilisateur et la gestion des tickets d'incidents, réparti de fait pour l'instant sur trois centres, en attendant une structuration régionale
- l'ENOC (EGEE Network Operation Center) , pour le suivi du réseau en liaison avec GEANT.

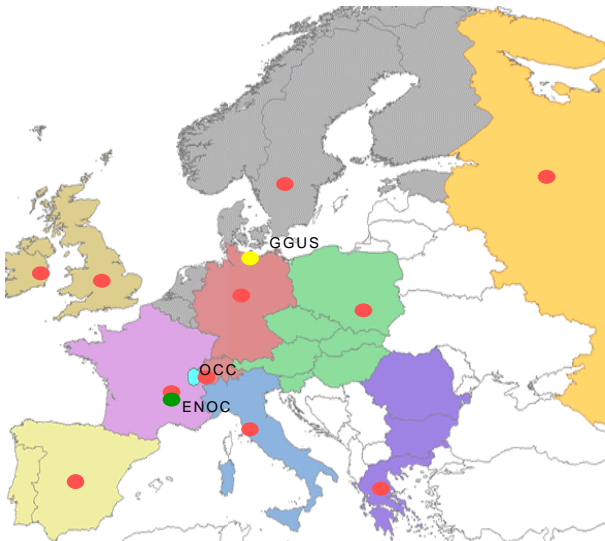


Figure 9 - Organisation de l'administration de EGEE

4.7.2 Métrologie

On ne peut exploiter des systèmes de grande taille sans avoir un minimum d'outils permettant de visualiser les activités, d'analyser les performances pour détecter les optimisations possibles.

Des outils sont disponibles depuis longtemps; nous citerons notamment l'adaptation de Ganglia (Figure 10), pour le monitoring de l'activité de grilles allant jusqu'à plusieurs dizaines de grappes:

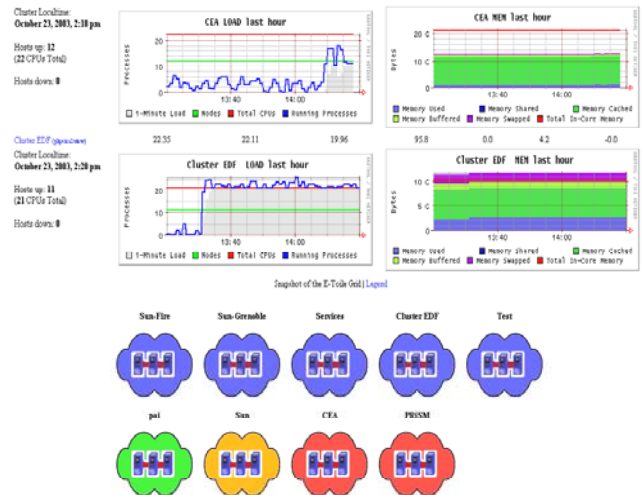


Figure 10 - Monitoring de la grille RNTL e-Toile

Cependant, le volume d'information à transférer et traiter devient sensible pour des grilles de très grande taille, et peut impacter leur utilisation.

Ces outils travaillent par observation périodique du statut de blocs de contrôle des systèmes, sans corrélation avec les applications. Or, l'étude du comportement de celles-ci sur grilles constitue un point majeur. Tout système digne de ce nom fournit des informations sur l'usage de ces ressources en fin de job. La constitution d'enregistrements -les Usage Records [8] - à partir de ces informations bénéficie d'un format standardisé par le Global Grid Forum (<http://www.gridforum.org/>). Leur intérêt est parfois mal perçu par certains universitaires, qui confondent la comptabilisation (accounting) de l'usage des ressources par unité de travail avec leur facturation.

Le collationnement des informations de monitoring et d'usage, et leur accès aisé constituent un point important pour l'analyse des performances et la maîtrise des systèmes de grilles. Là aussi, des standards existent: GMA (Grid Monitoring Architecture [10]), permettant d'envisager de nouveaux outils [7]).

5 Les applications concernées

Toutes les applications ne justifient pas l'usage de grilles, notamment celles existantes dont le code est difficile à paralléliser même à très gros grain, mais aussi de par les délais d'allocations des ressources et des latences inhérentes aux réseaux inter nœuds des grilles. Au delà de la mise à disposition de ressources plus importantes en permanence ou à la demande, des exemples montrent bien les domaines dans lesquels elles deviennent indispensables:

- l'étude d'interactions par couplage d'applications très consommatrices opérant sur des systèmes différents, typiquement le comportement de structures face à

l'injection de fluides, de plasmas ou soumis à des champs magnétiques

- le LHC, prochain 'collisionneur de particules' du CERN va produire 4*40 millions événements par seconde (15 Petaoctets sur 10⁹ fichiers/an) qui seront visualisés et analysés par 6000 physiciens dans le monde via des centres de calcul répartis sur quatre niveaux
- des applications chirurgicales avec visualisation 3D en temps réel de la pénétration d'instruments dans des corps humains découpés en tranches (chacune de 20 Ko à 400 Mo) vont passer à l'expérimentation en liaison avec des hôpitaux
- les volumétries de données à traiter dans beaucoup de sciences sont énormes, mais elles sont aussi partagées entre de multiples organismes et doivent être accédées au travers de métadonnées descriptives des ontologies, des droits d'accès, des protocoles propres à chaque organisme; ceci est courant dans certains domaines comme la climatologie, mais on doit songer à des applications interdisciplinaires telles que le data mining sur des données sociologiques ou économiques
- un exemple typique en biologie est celui de la recherche *in silico* de médicaments par l'analyse de ligands (molécule capable de se lier spécifiquement à une macromolécule réceptrice) avec l'annonce récente de l'analyse de l'interaction inhibitrice potentielle de 46 millions de ligands dans la malaria, traitements exécutés pendant un mois sur 1000 machines et générant plusieurs téraoctets de données sur des millions de fichiers²
- la notion de grille recouvre la coopération de serveurs, pas seulement affectés à du calcul mais pouvant exécuter des services spécialisés, localisés dans des organismes différents, comme par exemple des serveurs de vidéoconférence comme dans Access Grid (<http://www.accessgrid.org>) et d'autres projets de présentations interactives et vidéo à la demande
- nous évoluons de plus en plus vers une notion de laboratoire virtuel multi-organismes dans lequel les expérimentateurs disposent d'outils informatiques leur permettant de visualiser en temps réel et en 3D ou 4D les résultats de modélisations dynamiques d'une réalité complexe (c'est à dire avec des traitements interagissant sur des volumes de données distribuées très importants).

Des applications simples peuvent être réalisées aisément en mode maître-esclave avec un jeu de données propre à chaque exécution sur chacun des nœuds. Au delà, une compétence en 'gridification' des applications s'avère souhaitable pour adapter le code. Les techniques de 'gridification' les plus connues sont:

- conception des applications en maître-esclave (call "ex-subroutine")
- usage de RPC (Remote Procedure Calls) avec ou sans serveur intermédiaire
- découpage en gros grains d'applications patrimoniales par encapsulation des appels de sous-routines vers des services intermédiaires en CORBA par exemple
- subdivision en traitements associés à des sous-ensembles de données
- échanges entre applicatifs tournant sur des nœuds différents via des messages en MPI (Message Passing Interface) adapté aux grilles (MPICH-V)
- usage de langages adaptés (HPF, C*, C++, Java) et de bibliothèques de codes spécialisés, packages et solveurs gridifiés (NetSolve...) et intégrant le MPI (Scalpack...)
- écriture des nouvelles applications et réingénierie des applications patrimoniales en approche native objet C++, Java
- en orientation portail pour l'ergonomie, mais aussi pour gérer le flot du travail.

6 Utilisation en entreprise

Des freins existent pour l'adoption des grilles, tant dans le monde académique qu'en entreprises. Une des difficultés vient de l'adaptation des applications à la parallélisation d'exécution dans un premier temps, puis aux particularités des grilles avec l'introduction de délais au démarrage, des temps accrus de passage de messages entre tâches, la non persistance locale de données, etc.. Des démarches de "gridification" existent à cet effet. Une autre différence avec le monde de la recherche académique est dans l'usage plus fréquent de SGBD pour lesquels des problèmes peuvent surgir lorsqu'on pousse leur usage aux limites, par l'usage de bases de données distribuées sur des nœuds hétérogènes de grilles [3]. Mais généralement le SGBD et ses bases sont implantés sur des serveurs spécifiques assurant leur permanence et la performance d'accès, et non sur des nœuds de grille. Une étude fonctionnelle approfondie des transactions distribuées permet de s'assurer de leur caractère ACID (atomicité, consistance, isolation, durabilité) dans tous les cas, et de préparer des plans de requêtes avec réservation de serveurs spécifiques lors d'opérations -jointures notamment- manipulant des sous-ensembles conséquents.

En entreprise, les grilles se mettent surtout en place dans de très grandes sociétés à haute technologie pour la modélisation, la simulation et la visualisation 3D dans l'automobile, l'aéronautique, le spatial, le recherche pétrolière et le nucléaire. Les autres projets couvrent le calcul intensif et une meilleure gestion du parc informatique avec son obsolescence rapide. Le Desktop Grid se généralise rapidement dans la pharmacie, la biologie, la modélisation moléculaire et la chimie. En

² cf <http://public.eu-egec.org/pr/>

finance et bourse, on a pu observer un abaissement drastique des traitements dans l'analyse de risque dans la finance et la bourse. Un cas type est celui de la CGG qui dispose de 20000 PC multiprocesseurs réunis en clusters sur plusieurs continents pour exécuter ses applications de sismique, et souhaite passer en mode grille afin d'optimiser dynamiquement la charge entre eux et constituer un ensemble virtuel de traitement et de stockage partagé entre clients avec modification en tant que de besoin des ressources affectés.

Il ne faut pas croire les grilles réservées aux grandes entreprises. Plusieurs fédérations de PME s'y intéressent pour le partage dynamique de leurs équipements, données techniques et codes de calcul, ou en vue d'utilisation en mode informatique à la demande ou ASP avec des fournisseurs externes, pour des coûts actuellement hors de leur portée.

7 Les principaux acteurs en France

Nous avons listé Figure 11 des produits à examiner pour se lancer dans les grilles:

Entreprise / Organisme	Produit	Origine	Technologie
Avaki	Legion (Compute & Data)	USA	Mixte
CNRS/LRI	XtremWeb	F	Desktop Grid
CNRS/LAAS+ Alinka SA	Aroma	F	Outils d'administration
CNRS/INRIA/CEA/EDF/CS/SUN	e-Toile	F	Massive Grid
Data Synapse	LiveCluster	USA	Desktop Grid
Entropia	DC Grids	USA	Desktop Grid
GridSystems	Inner Grid	SP	Desktop Grid
GridXpert	Synergie	F -> US	Outils d'administration
Icatis	ComputeMode	F	Extension Cluster>PC
HP	Utility Data Center	USA	Massive Grid (1e gén.)
IBM	Pas de produit propre	USA	
Meshtechologies	OfficeGrid	DK	Desktop Grid
Pallas->Intel	Unicore	D->USA	Massive Grid (1e gén.)
Platform	LSF/Symphony	USA	Mixte
Sun	Sun Grid Engine	USA	Massive Grid (1e gén.)
Université d'Argonne	Globus	USA	Massive Grid
Université de Berkeley	Boinc	USA	Desktop Grid
Université du Wisconsin	Condor	USA	Massive Grid
United Devices	Grid MP	USA	Desktop Grid
W2G technologies	Web2grid	F	Virtualisation (V V M)

Figure 11 - Quelques fournisseurs et produits à considérer

8 Conclusion

Les technologies vont encore évoluer, mais les premières générations de grilles permettent déjà de résoudre des problématiques, jusque là insolubles, de prise en compte dans des délais et des coûts raisonnables de traitements coopératifs ou intensifs sur des données massives distribuées.

Les grilles informatiques commencent à se généraliser et vont avoir un impact fort sur les profils d'administrateurs systèmes et réseaux; il est donc temps pour eux de se former aux concepts pour disposer d'une vision globale avant d'étudier, de devoir implanter, tel ou tel produit.

Bibliographie

ARCHITECTURES

- [1] Foster Ian, Kesselman Carl, Nick Jeffrey M., Tuecke Steven: The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, *Global Grid Forum* (first draft: 22nd June 2002 et suivants à : <http://www.gridforum.org/ogsawg/>)
- [2] Georges Fedack: Le projet XtremWeb à l'INRIA Futurs, *2ème Congrès National de Mathématiques Appliquées et Industrielles*, 23 mai - 27 mai 2005
- [3] Sushant Goel, Hema Sharda, David Taniar: Atomic commitment and resilience in grid database systems, *International Journal Grid and Utility Computing*, Vol 1, No 1, 2005
- [4] Web site gLite (Lighthouse Middleware for Grid Computing): <http://glite.web.cern.ch/glite/>

MODELES ECONOMIQUES

- [5] Buyya Rajkumar: Economic-based Distributed Resource Management and Scheduling for Grid Computing, PhD Thesis, *Monash University*, Melbourne, Australia, April 12, 2002.
- [6] J. Nabrzyski, J. Schopf and J. Weglarz, Kluwer: Grid Resource Commercialization: Economic Engineering and Delivery Scenarios, *Grid Resource Management: State of the Art and Research Issues*, 2004 (<http://www.zurich.ibm.com/grideconomics/refs.html>)

METROLOGIE

- [7] Sergio Andreozzi, Natascia De Bortoli, Sergio Fantinel, Antonia Ghiselli, Gian Luca Rubini, Gennaro Tortone, Maria Cristina Vistoli, GridICE: a Monitoring Service for Grid Systems, *Future Generation Computer Systems Journal*, Elsevier, 21(4):559-571
- [8] John Gordon: Accounting 'the last A' and Record Usage Service, *e-IRG Amsterdam workshop*, May 13th 2005
- [9] Jason, Lee, Dan Gunter, Martin Stoufer, Brian Tierney: Monitoring Data Archives for Grid Environments, Conference on High Performance Networking and Computing archive, *Proceedings of the 2002 ACM/IEEE conference on Supercomputing* p1-10
- [10] B. Tierney, R. Aydt, D. Gunter, W. Smith, M. Swany, V. Taylor, R. Wolski: A Grid Monitoring Architecture, *Global Grid Forum GGF Performance Working Group*, March 2000, Revised 16-January-2002
- [11] Serafeim Zanicolas, Rizos Sakellariou: A taxonomy of grid monitoring systems, *Elsevier Future Generation Computer Systems* 21 (2005) 163-188

SECURITE ET CONFIDENTIALITE

- [12] Collberg C, Thomborson C, Low D: A taxonomy of obfuscating transformations (technical report), *University of Auckland*, July 1997
- [13] Germain-Renaud Cecile: Contributions à la modélisation et l'optimisation de systèmes de calcul à grande échelle, HDR soutenue le 7 septembre 2005, <http://www.lri.fr/~cecile/RAPH/germainHdr.pdf>
- [14] Jingmin He, Min Wang: Cryptography and Relational Database Management Systems, *2001 International Database Engineering & Applications Symposium*, p. 0273
- [15] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, Yirong Xu: hippocratic Databases, *28th Int'l Conf. on Very Large Databases (VLDB)*, Hong Kong

OUVRAGES

- [16] Baude Françoise: Calcul réparti à grande échelle - Metacomputing, 2002 ISBN:2-7462-0472-X *Hermès-Lavoisier*
- [17] Foster & Kesselman: The Grid: Blueprint for a New Computing Infrastructure, *Morgan Kaufmann*, 2003 (2ème édition en 2004 sous le titre GRID2)
- [18] Fox, Berman, Hey: Grid Computing: Making the Global Infrastructure a Reality, *Wiley and Sons*, 2003
- [19] Joshy Joseph, Craig Fellenstein: Grid Computing, *Prentice Hall PTR* 2003
- [20] Marcel Soberman: Les grilles informatiques, ISBN: 2-7462-0633-1, *Hermès - Librairie Lavoisier*, 2003