

=====

DGIS : Approche distribuée pour une optimisation de la découverte de ressources dans les grilles

Cheikhou Thiam	Ibrahima Niang	Samba Ndiaye	Yahya Slimani
Dépt. de Maths-Info	Dépt. de Maths-Info	Dépt. de Maths-Info	Dépt info
Univ. CA Dakar	Univ. CA Dakar	Univ. CA Dakar	Univ. El Manar Tunis
BP 5005 – Dakar – Fann	BP 5005 – Dakar – Fann	BP 5005 – Dakar – Fann	
SENEGAL	SENEGAL	SENEGAL	TUNISIE
ethiam@univ-thies.sn	iniang@ucad.sn	samba.ndiaye@ucad.edu.sn	yahya.slimani@fst.rnu.tn

=====

RÉSUMÉ. La gestion du système d'information de Globus, matérialisé par son MDS (Metacomputing Directory Service) avec son seueur GIIS (Grid Index Information Service), influe fortement sur les performances d'une grille informatique. Plusieurs solutions d'optimisation basées sur les réseaux structurés ont été proposées. Cependant, elles sont peu adaptées aux grandes réplifications et aux environnements très dynamiques. En tenant compte de ces insuffisances, nous proposons une approche distribuée de recherche de ressources nommée DGIS (Distibuted Grid Index Server), basée sur les réseaux non structurés qui sont très adaptés aux environnements dynamiques comme les grilles.

ABSTRACT. The management of Globus information system, materialized by its MDS (Metacomputing Directory Service) with its GIIS server (Grid Index Information Service), strongly influences the performances of data-processing grid. Several optimized solutions based on structured networks are proposed. However, they are not adapted for large replications and very dynamic environments. Taking account these insufficiencies, we propose a distributed approach named DGIS (Distibuted Grid Index Server) for resources discovery, based on not structured networks which are very adapted to the dynamic environments like grid computing.

MOTS-CLÉS : Grille, Annuaire, réplification, système d'information, P2P

KEYWORDS: Grid, Annuaire, replication, Information system, P2P

=====

=====

1. Introduction

La gestion des ressources dans les grilles est un problème fondamental qui met en œuvre un certain nombre de structures de données et de services dans le cadre d'un middleware tel que GLOBUS. Ces structures de données et ces services constituent le Système d'Information d'une grille, qui est matérialisé par le MDS (Metacomputing Directory Service) [6].

Beaucoup de services de découverte de ressource intégrés dans les middlewares adoptent une approche centralisée ou hiérarchique. Cependant, l'évolution de l'architecture nécessite d'avoir des systèmes respectant le passage à l'échelle avec des milliers de nœuds. Actuellement, la plupart des solutions existantes dans les systèmes P2P (Peer-To-Peer) reposent sur les réseaux structurés. Ces réseaux associent la localisation des informations à la topologie du réseau et sont particulièrement adaptés que pour retrouver des informations peu répliquées. La structuration du réseau a un coût en termes de bande passante utilisée et demande une certaine stabilité des nœuds. En plus, au niveau de la table de hachage distribué (DHT) [1], les requêtes sont limitées aux ressources dont l'identifiant est connu. En plus des valeurs de hachage très différentes peuvent avoir des clés très proches ce qui justifie le coût élevé pour répondre à des requêtes approchées ou portant sur un intervalle.

Dans cet article, nous proposons une nouvelle approche de structuration des informations et de recherche des ressources de la grille basée sur les arbres AVL (arbre binaire de recherche équilibré) [2]. Cette approche optimise le temps de recherche quand la taille de la grille augmente. L'article est organisé comme suit : la section 2 présente quelques solutions de distribution de l'annuaire. Dans la section 3 nous présentons notre approche de distribution de l'annuaire nommée DGIS. Nous proposons une validation théorique et expérimentale de la section 4.

2. Background

Dans [5], il a été proposé une solution distribuée du MDS nommée SAMGrid qui repose sur la topologie Chord [8] avec des IS (Information Service) à chaque nœud. Quand un nœud a besoin d'un fichier, l'IS consulte d'abord l'anneau Chord; si le fichier n'existe pas dans l'anneau la requête se poursuit au niveau d'un serveur central. L'article [7] présente une approche basée sur un anneau Chord formé de pairs (points noir) et d'informations envoyées par les sites (points bleus). Chaque site est lié à un pair par le « courtier de grille ». Les courtiers créent un anneau Chord, qui maintient collectivement l'index logique pour faciliter un procédé de découverte décentralisé de ressource. La solution DGRID présentée dans [4] est organisée en domaines administratifs. Chaque domaine administratif stocke ses propres données. Une fonction de hachage (ex : SHA-1) est appliquée au nom du domaine pour obtenir l'identifiant. La

Approche distribuée pour une optimisation de la découverte de ressources dans les grilles 3

même opération est effectuée pour obtenir la clé de chaque type de ressource. Les approches étudiées reposent sur des réseaux structurés basés sur Chord. Le mécanisme de recherche est le même pour ces trois solutions avec une complexité de $O(\log N)$. Cependant, les trois solutions sont basées sur des réseaux structurés.

3. Approche de distribution et architecture de DGIS

3.1 Architecture de DGIS et organisation des ressources

Chaque site dispose des composants suivants (voir Figure 1) : (i) les nœuds simples (CE : élément de calcul, SE : élément de stockage), (ii) les super-nœuds. Ils représentent les GIIS (Grid Index Information Service) des sites. Tous les nœuds d'un site annoncent leurs ressources à leur GIIS local. Les différents sites de la grille sont reliés par les GIIS. La communication inter-site se fait entre les GIIS, (iii) un annuaire local sous forme d'un arbre AVL local, qui synthétise l'ensemble des ressources d'un site, (iv) une table des voisins, (v) un cache, représenté par une table, qui contient un ensemble de « connaissances » sur les ressources des autres sites. Le contenu de ce cache est dynamique. En fait, ce cache permettra à un site d'avoir une idée sur les ressources des autres sites.

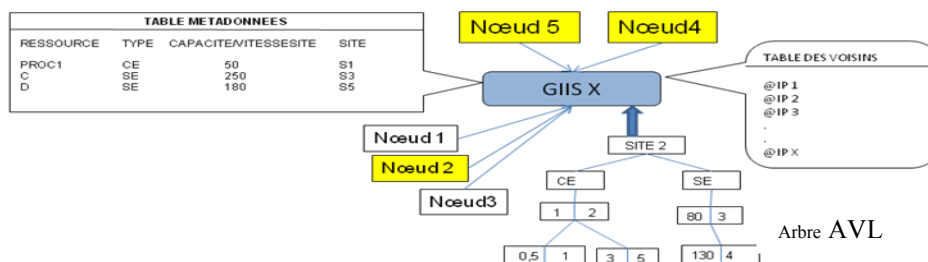


Figure 1. Composantes d'un site

Les différents GIIS sont reliés entre eux à travers un réseau non structuré. Ce qui leur permettra de communiquer entre eux pour assurer une gestion de l'ensemble des ressources d'une grille.

La figure 2 illustre un exemple de grille de DGIS (réseau de GIIS)

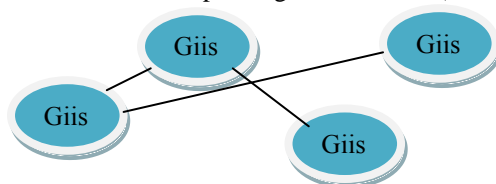


Figure 2. Exemple architecture d'une grille DGIS (réseau non structuré)

3.3 Fonctionnement de DGIS

3.3.1 Ajout et suppression d'un nœud

L'ajout d'un nœud se fait au niveau local. Ce nouveau nœud s'enregistre auprès du GIIS du site auquel il appartient et ses ressources sont insérées dans l'arbre AVL local. L'insertion des ressources est celle utilisée par les arbres AVL. L'opération de suppression est similaire à l'insertion, mais avec quelques petites différences. La suppression d'une ressource est identique à celle d'un nœud dans un arbre AVL. Néanmoins, l'arbre devra être rééquilibré en cas de déséquilibre, afin de préserver les propriétés d'un arbre AVL.

3.3.2 La recherche de ressource

Toute recherche commence par une identification du type de la requête, CRQ (Computing Request Query) ou SRQ (Storage Request Query) dont la structure est représentée $Req(i, p, t, l)$. Une requête sera caractérisée par son identifiant k , l'adresse du GIIS initiateur de la recherche p , du texte (contenu) de la requête t et d'un TTL l .

3.3.2.1 Recherche locale

La recherche se déroule exactement de la même manière que pour un arbre AVL avec une complexité égale à $O(\log N)$. Cependant la recherche n'est plus binaire dans le cas d'une recherche d'espace disque non contigu; pour ce cas l'arbre est parcouru jusqu'à ce qu'on obtienne l'espace demandé ou qu'on parcourt tout l'arbre.

3.3.2.2 Recherche dans la grille

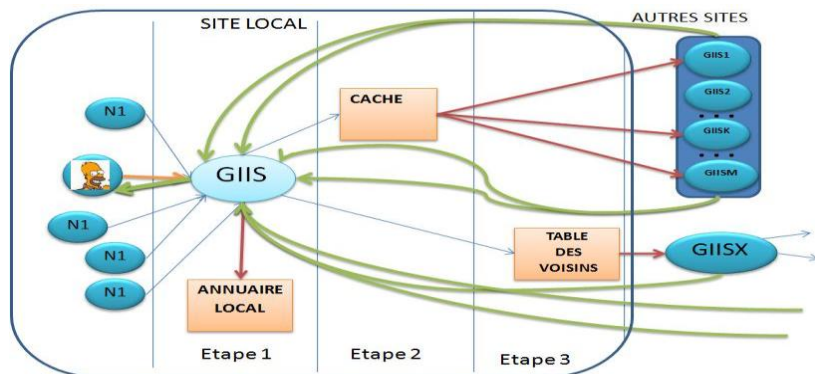


Figure 3 : Les différentes étapes pour la localisation de ressource

La méthode que nous proposons consiste à interroger les nœuds qui ont le plus de chance de répondre aux requêtes des utilisateurs, en utilisant les réponses de requêtes déjà traitées. Cette connaissance sur les requêtes passées (historique) permettra : (i) d'interroger moins de voisins et donc de minimiser l'utilisation de la bande passante ; et

Approche distribuée pour une optimisation de la découverte de ressources dans les grilles 5

(ii) de diminuer la latence tout en ayant le maximum de réponses. Lorsqu'un nœud effectue une requête de découverte de ressources, la recherche commence par l'annuaire local. Si le nombre de réponses positives n'est pas suffisant, alors le cache est consulté et si des ressources présentes dans le cache correspondent aux critères demandés, elles sont contactées en priorité. Ensuite, si le nombre de réponses reçu est toujours insuffisant, la requête est renvoyée au voisin qui a le ratio le plus élevé (figure 5).

L'utilisation d'un cache permet d'économiser de la bande passante. Pour que ce cache ait un intérêt dans notre contexte d'utilisation, il faut lui adjoindre une politique de gestion efficace. Nous avons introduit des opérations fondamentales pour une bonne gestion du cache :: (i) le nœud dont la date d'allocation est la plus récente est supprimé ; (ii) si la date d'allocation ne permet pas de faire la différence entre les nœuds présents dans le cache, c'est l'entrée la plus ancienne qui est supprimée ; (iii) si la date d'insertion ne permet pas non plus de faire la différence entre nœuds, alors une entrée sera supprimée au hasard.

4. Validation théorique et expérimentale

4.1 Validation théorique

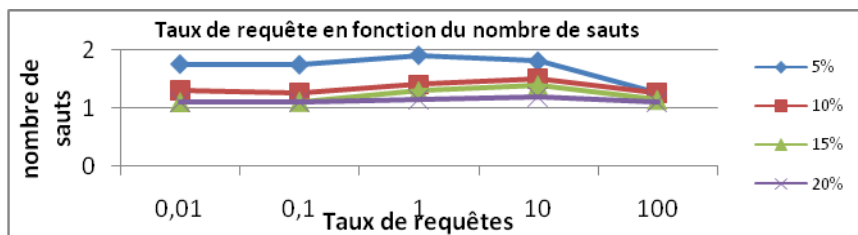
Etant donné que chaque site gère ses propres ressources, les opérations d'ajout et de suppression de ressources se font localement. Relativement à la répartition de ressources, le modèle DGRID [8] utilise un réseau structuré. Les utilisateurs de notre système pourront rechercher des ressources disponibles sur les différents sites d'une grille. Pour définir les caractéristiques de notre solution, nous l'avons comparé avec celle de DGRID.

La recherche locale au niveau de chaque site est identique à celle des arbre où les opérations sont en $O(\log_2 N)$. Recherche globale. La recherche a une complexité moyenne de $O(\log_2 N)$. Cette comparaison est synthétisée dans le tableau 4.2.

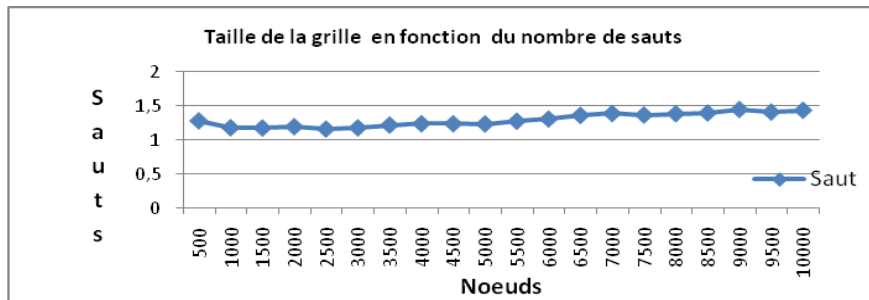
Réseau	DGRID	DGIS
TYPE DE RESEAU	Structure	Non structure
RESSOURCE	Enregistrement sur un	Au niveau du GIS
REPRESENTATION DES RESSOURCES	Liste. Utilisation de SHA1-hashing	Arbre AVL (recherche dichotomique)
RECHERCHE	Basée sur la valeur de l'ID de la ressource. $O(\log_2 Y)$, Y est le nombre ressources dans	Multicritères. $O(\log N)$ en local, N nombre de ressource local
FORTE REPLICATION	Pas adaptée	Adaptée

Tableau 1. Etude Comparative de DGIS et DGR/D**4.2 Validation expérimentale**

Nous avons implémenté dans le simulateur OverSim [4]. Les requêtes de simulation seront générées à des intervalles de 100, 10, 1, 0.1, et 0.01 secondes. Nous avons utilisé un nombre de nœuds qui varie entre 500 et 10000. Dans cette première série d'expérimentations, nous mesurons la surcharge de la grille en augmentant le nombre de requêtes. A travers cette série de tests, nous voulons vérifier si le nombre m de requêtes a un effet ou pas sur le comportement du système.

**Figure 4. effets du nombre de nœuds contenant la même ressource sur les performances**

La figure 4 montre que la croissance de la grille n'a pas une grande influence sur le nombre de sauts nécessaires pour retrouver une ressource quand la fréquence de cette dernière augmente et donc sur les performances du système. La structure en arbre AVL des annuaires locaux et la rapidité de traitement des requêtes au niveau d'un site ont contribué à l'amélioration de la recherche. Ce résultat est très important d'autant plus que cette fréquence pose un problème dans les grilles structurées utilisant des DHT au cours des recherches, les valeurs semblables pouvant avoir des clés très différentes.

4.3 Evolutions du nombre sauts et de la latence**Figure 5. Evolutions du nombre de sauts en fonction du nombre de nœuds**

Approche distribuée pour une optimisation de la découverte de ressources dans les grilles 7

DGIS supporte bien le facteur d'échelle. A partir de ces figures 5, nous voyons que les simulations menées montrent que la longueur moyenne d'un chemin utilisé pour acheminer une requête évolue en $O(\log N)$.

4.4 Comparaison de DGIS et DGRID

Dans la deuxième série d'expériences, nous proposons de mesurer le nombre de sauts et la latence. Nous comparons ensuite les résultats obtenus avec ceux de DGRID.

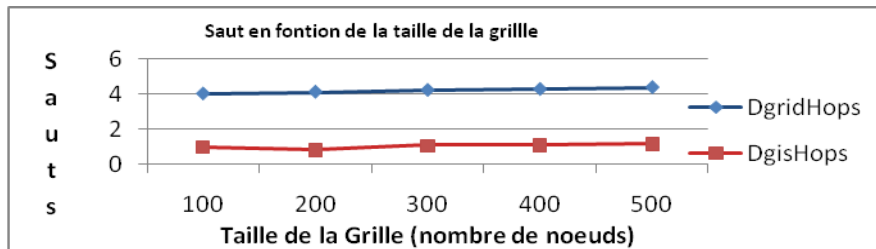


Figure 6. Performances de DGIS vs DGRID

La figure 6 représente l'évolution du nombre de saut en fonction de la taille de la grille. Le nombre de sauts pour DGIS est nettement inférieur à celle de DGRID.

6. Conclusion

L'utilisation d'une grille non structurée rend le système de recherche plus flexible. En effet, les opérations d'ajout et de suppression de ressources dans la grille se résument en une opération locale d'insertion dans un arbre AVL. Nous avons proposé une optimisation du protocole de marches aléatoires afin d'en augmenter l'efficacité. Cette solution, distribuée pour la recherche de ressources dans les grilles, repose aussi sur l'utilisation optimale du cache. Cette architecture permet de découvrir des ressources, dans une grille, suivant les critères qui représentent les pré-requis matériels et logiciels pour l'exécution d'une application. Les résultats de nos simulations montrent que notre solution s'adapte mieux quand la taille de la grille augmente. Le nombre de sauts est nettement inférieur à celui de DGRID. En plus notre solution supporte bien le facteur d'échelle.

7. Bibliographie et Biographie

7.1. Bibliographie

- [1] ABE Hirotake; “DHT : Distributed Hash Table-Infrastructural Technology for Pure Peer-to-Peer Systems”. J. Title;Konpyuta Sofutowea VOL.23;NO.1;PAGE.1-14(2006).
- [2] Adel'son-Vel'skiĭ (G.M.) et Landis (E. M.). “An algorithm for the organization of information”. Soviet Mathematics Doklady, 3 : 1259–1263, 1962.
- [3] I. Baumgart, B. Heep, and S. Krause, “OverSim: A flexible overlay network simulation framework,” in Proceedings of 10th IEEE Global Internet Symp. (GI '07) in conjunction with IEEE INFOCOM 2007, Anchorage, AK, USA, May 2007, pp. 79–84.
- [4] March,V. Teo, Y.M. and Wang, X. DGRID : “A DHT-Based Resource Indexing and Discovery Scheme for Computational Grids”. In Proc. Fifth Australasian Symposium on Grid Computing and e-Research (AusGrid 2007), Ballarat, Australia. CRPIT, 68. Coddington, P. and Wendelborn, A., Eds., ACS. 41-48.
- [5] Matthew Leslie, Sinisa Veseli : “SAMGrid Peer-to-Peer Information Service”. Oxford University Computing Laboratory Fermi National Accelerator Laboratory. Presentations at the Computing in High Energy and Nuclear Physics 13-17 February 2006, T.I.F.R. Mumbai, India
- [6] Joseph Bester, Ian Foster, Carl Kesselman, Jean Tedesco et Steven Tuecke. GASS: “A Data Movement and Access Service for Wide Area Computing Systems”. In 6th Workshop on Input/Output in Parallel and Distributed Systems (IOPADS), pages 78–88. ACM Press, Atlanta, GA, mai 1999
- [7] Rajiv Ranjan, Lipo Chan, A. Harwood, Shanika K., and Rajkumar B., “Decentralised Resource Discovery Service for Large Scale Federated Grids”, Proceedings of the 3rd IEEE International Conference on e-Science and Grid Computing (e-Science 2007, IEEE CS Press, Los Alamitos, CA, USA), Dec. 10-13, 2007, Bangalore, India.
- [8] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan : “Chord: A scalable peer-to-peer lookup service for internet applications“. In Proc. SIGCOMM '01, pages 149–160, New York, NY, USA, 2001. ACM Press.

7.2. Biographie

Cheikhou Thiam : étudiante en thèse à l'école doctorale Mathématique-informatique de Dakar (Sénégal) ; domaine de recherche : QoS, optimisation dans les réseaux P2P et WSN

Ibrahima Niang : Dr. en informatique, enseignant chercheur au dept de Mathématique-Informatique à l'université Cheikh Anta Diop de Dakar (Sénégal). Domaine de recherche : QoS, Sécurité et Mobilité dans les réseaux P2P et sans fil.

Samba NDiaye : Dr. en informatique, enseignant chercheur au dept de Mathématique-Informatique à l'université Cheikh Anta Diop de Dakar (Sénégal). Domaine de recherche : Base de données et systèmes distribués.

Yahya Slimani : Professeur en informatique à l'université El Manar Tunis (Tunisie). Domaine de recherche : datamining et fouille de données dans les réseaux P2P et grille de calcul.