

Répartition dynamique de données massives de biodiversité à la demande

CNRIA 2015

Ndiouma Bame*, Hubert Naacke*, Idrissa Sarr† and Samba Ndiaye†

* Sorbonne Universités, UPMC Univ Paris 06

LIP6 Laboratory Paris, France

prenom.nom@lip6.fr

† Département de mathématique-informatique

Université Cheikh Anta Diop, Dakar, SENEGAL

prenom.nom@ucad.edu.sn

Résumé—La problématique du ‘Big Data’ fait parti des défis informatiques majeurs de la décennie en cours. La quantité des données produites augmente constamment et rend leur traitement de plus en plus difficile à gérer avec les outils actuels. L’organisation et l’interrogation de ces masses de données sont essentielles pour élargir les connaissances du domaine y afférent. Dans cet article, nous proposons une solution de répartition dynamique des masses de données de biodiversité pour adapter la distribution des données en fonction de leur structure hiérarchique et des demandes versatiles des utilisateurs. Ces situations sont caractérisées par des charges et des popularités de données disparates et fluctuantes à travers les sites. Or, les solutions actuelles ne s’adaptent pas dynamiquement à ce type de situation. Nous avons implémenté notre solution dans un cluster de 200 cœurs. Les résultats de la validation expérimentale montrent l’adaptation dynamique de notre solution pour l’organisation des données en fonction des demandes et des charges des sites.

I. INTRODUCTION

Dans la décennie en cours de nombreux domaines font face à un ‘déluge’ de données sans précédent. La quantité des données produites augmente constamment et rend leur traitement de plus en plus difficile à gérer avec les outils actuels. L’organisation et l’interrogation de ces nouvelles masses de données sont essentiels pour élargir les connaissances du domaine y afférent. Ceci est particulièrement crucial dans des domaines tels que les média sociaux, la génomique, la climatologie, les réseaux énergétiques complexes, l’astronomie, l’écologie et la biodiversité. Parmi ces domaines, nous nous intéressons à celui de la biodiversité pour lequel le GBIF fédère et partage les données de biodiversité produites par de nombreux fournisseurs à l’échelle mondiale. Toutefois, avec un nombre croissant de fournisseurs qui ajoutent de nouvelles données et d’utilisateurs qui expriment de nouveaux besoins d’interrogation, l’accès aux données du GBIF pose un double problème d’expressivité et d’efficacité difficile à résoudre. Plus généralement, la gestion de masses de données nécessite l’utilisation de nouveaux systèmes de gestion de données disposant de méthodes d’accès performantes. Un aspect particulièrement difficile à prendre en compte est le comportement très versatile des utilisateurs. Cela génère des demandes très fluctuantes : la charge est variable en nombre

de requêtes, certaines données sont plus populaires (fréquemment demandées) que d’autres, la popularité est elle-même fluctuante (une données n’est populaire que pendant une durée limitée). Or, les solutions actuelles [1], [2], [3], [4], [5], [6] n’ont pas été conçues pour s’adapter dynamiquement à ce type de situation. En effet, même s’il existe quelques solutions d’adaptation dynamique du placement des données à travers les sites, celles ne considèrent pas tous les paramètres que nous jugeons être indispensable pour un placement dynamique de données à large échelle efficace. A la suite de nos travaux précédents [7], [8], [9], nous proposons dans cet article une approche de répartition dynamique des données de biodiversité pour garantir un temps de réponse raisonnable pour les requêtes et le passage à l’échelle des données et des traitements. En effet, la politique de répartition des données à travers un environnement distribué joue un rôle essentiel pour la garantie de performances du système vis-à-vis des applications. Dans cet article, nous présentons notre approche de répartition dynamique des données de biodiversité à travers l’architecture décentralisée [8]. Elle est basée sur la structure hiérarchique des données de biodiversité et les spécificités des requêtes d’analyse qui les exploitent. Nous complétons notre politique de répartition dynamique des données avec une stratégie de remplacement qui a pour objectif d’optimiser les ressources de stockage tout en garantissant la disponibilité des données. La section 2 présente notre approche de fragmentation à la demande. La section 3 présente notre stratégie de placement et de répartition dynamiques des fragments à travers les sites. Dans la section 4, nous présentons notre stratégie de remplacement. La section 5 détaille notre validation expérimentale et la section 6 conclue.

II. FRAGMENTATION MULTIDIMENSIONNELLE DES DONNÉES À LA DEMANDE

Les données et les requêtes de biodiversité ont des spécificités qui peuvent être exploitées pour la fragmentation. En effet, les données de biodiversité présente une structure hiérarchique selon plusieurs dimensions. Or, les requêtes d’analyse de biodiversité ont généralement des prédicats qui portent sur ces dimensions. Nous détaillons ces propriétés des données et des

requêtes avant de présenter notre approche de fragmentation des données à la demande.

A. Organisation hiérarchique des données de biodiversité

Les données de biodiversité ont la particularité d'avoir une structure hiérarchique selon plusieurs dimensions. En effet, les dimensions de la classification taxinomique, géographique et temporelle, qui décrivent une observation (occurrence), présentent chacune une structure hiérarchique. En outre, ces dimensions sont généralement utilisées dans les analyses de données de biodiversité. Les dimensions taxinomiques et géo-spatiales sont les plus fréquentes dans les requêtes d'analyse. La taxinomie permet de décrire les organismes vivants et de les regrouper en entités appelées taxons. Il permet d'identifier, de nommer et de classer les taxons et de les reconnaître via des clés de détermination dichotomiques. La taxinomie fixe les critères et les règles de classification des taxons. Une observation ou occurrence consiste à une instanciation d'une espèce (niveau le plus bas de la hiérarchie). La description taxinomique d'une occurrence, consiste à renseigner toutes ses propriétés taxinomiques. Ces dernières correspondent à son nom et ceux de ses parents successifs. C'est ainsi que les données de biodiversité qui consistent aux descriptions des occurrences présentent une structure hiérarchique sur la dimension taxinomique. Les données géo-spatiales (ou géographiques) sont essentielles à la description des occurrences de biodiversité pour les exploitations des données dont elles peuvent faire l'objet. La description géo-spatiale d'une observation consiste à décrire l'endroit où cette observation est effectuée. La description est autant plus complète qu'elle tend vers un point unique identifié pour ses coordonnées géoréférentielles (latitude et longitude). Ce point d'observation est contenu dans une zone caractérisée par des limites géographiques. Partant du constat que le monde entier peut être considéré comme une zone géographique qui peut être divisée en plusieurs cellules (ou sous-zones) disjointes, on conclut que la description géo-spatiale présente une organisation hiérarchique. Le géoréférencement par les coordonnées latitude et longitude du point d'observation peut suffire pour décrire une observation géographique. Cependant lors de l'analyse des données, on arrive souvent à faire des regroupements d'un ensemble d'observation d'une même zone. On peut arriver aussi à élargir ou réduire la zone d'observation. D'où l'intérêt d'avoir une organisation hiérarchique de la description géographique.

B. Spécificités de requêtes d'analyse

Les cas d'usage de données de biodiversité disponibles au www.gbif.org/newsroom/uses, nous permettent de constater que les principes des analyses faites sur les données de biodiversité, ont en commun certaines dimensions interrogées. Ces dimensions concernent la dimension taxinomique pour connaître les taxons à analyser et la dimension géographique pour cadrer la zone de l'étude. Comme exemple, on peut citer le calcul de la cooccurrence de deux espèces (ou genre) dans une région (ou un pays), la distribution des éléphants

en Afrique, l'abondance des moustiques dans les régions tropicales, etc. Cette particularité des requêtes d'analyse de biodiversité se traduit par des prédicats portant sur les dimensions taxinomiques et géographiques. Ce qui peut jouer un rôle important dans le partitionnement des données. C'est ainsi que nous utilisons cette particularité des requêtes que nous combinons avec la structure hiérarchique des données pour fragmenter les données.

C. Définition des fragments à la demande

Nous fragmentons les données de biodiversité en fonction des dimensions hiérarchiques taxinomiques et géographiques. Pour cela, on suppose qu'une requête dispose de prédicats sur ces dimensions. De ce fait, le fragment est défini par la conjonction des prédicats sur ces deux dimensions : un prédicat de sélection du taxon analysé et un prédicat de la zone d'étude. Ainsi, un fragment est défini par la conjonction de nœuds dans la hiérarchie des dimensions taxinomiques et géo-spatiales. Cette structure hiérarchique des fragments permet de constater une relation de spécialisation/généralisation entre des fragments. Ce qui fait que bien que deux fragments n'aient pas les mêmes prédicats de définition, ils peuvent avoir une relation d'inclusion où le fragment le plus large inclut le plus petit. Ce qui favorise la réutilisation des fragments pour des requêtes qui ne les invoquent pas directement dans leurs prédicats. Un fragment est identifié par deux parties conjointes : nœud de la structure taxinomique et nœud de la structure géo-spatiale. Un nœud est défini par la concaténation de l'identifiant de son parent direct (supérieur hiérarchique) et son identifiant dans son groupe (éléments issues directement du même parent).

III. PLACEMENT ET RÉPLICATION DYNAMIQUES DES DONNÉES À LA DEMANDE BASÉES SUR UN COÛT

Le placement et la réplication de données ont pour objectif de garantir la disponibilité des données en réduisant les coûts de stockage et de transfert. Le problème principal de l'allocation de données est de trouver le schéma de placement qui optimise les performances du système avec un coût acceptable. Notre stratégie de placement des données distingue les deux phases du processus de répartition à savoir le placement initial (ou création) de fragments et la réplication de données. Le placement consiste à l'allocation des fragments issus du processus de fragmentation des données aux sites locaux et la réplication correspond à la duplication de certains fragments à travers les sites locaux. L'emplacement d'un fragment et/ou d'une nouvelle réplique sont décidées par le processus d'optimisation qui est basé sur un modèle de coût dynamique.

A. Création des fragments à la demande

Cette étape consiste à une réplication partielle à la demande ou de caching des données du GBIF vers les sites locaux. Les fragments sont déterminés grâce aux prédicats taxinomiques et/ou géographiques de la requête. De ce fait, les bases locales sont alimentées au fur et à mesure que les requêtes invoquent des fragments manquants (qui n'existent pas sur un site local).

D'où le concept de placement à la demande. Le site devant héberger chaque fragment est déterminé par l'optimisation basée sur le coût de la requête. Lorsqu'un fragment existe déjà dans un site local, alors ce site est utilisé en remplacement du portail GBIF pour la lecture ou le traitement du fragment. Ce placement à la demande, permet de minimiser les coûts de stockage et de transfert des données tout en favorisant le parallélisme des accès et des traitements à travers les sites locaux. Un processus de rafraîchissement des fragments est utilisé pour maintenir la cohérence des fragments locaux par rapport à la base complète du GBIF.

B. Réplication dynamique à la demande basée sur l'optimisation

Ayant comme objectif de traiter la plupart des requêtes dans délais impartis, la réplication d'un fragment sollicité qui se trouve sur un site surchargé peut jouer un rôle important. Puisque la charge d'un site est dynamique, notre approche d'optimisation basée sur le coût, coordonne les décisions de répliquions de fragments entre les sites locaux. L'objectif est de placer une réplique du fragment sur un site qui convient pour garantir un temps de réponse borné pour les requêtes qui interrogent le fragment. Ces nouvelles répliques pourront être utilisées lors des traitements des requêtes suivantes.

C. Optimisation de requête basée sur un modèle de coût dynamique

La stratégie d'optimisation de requête permet de trouver un plan d'exécution de la requête en fonction des paramètres qui sont pris en compte. Les paramètres d'optimisation considérés dans notre approche d'optimisation sont : la charge de chaque site, les localisations des répliques de chaque fragments invoqués, les capacités de calcul et de stockage de chaque site, les capacités des liens de communication entre les sites, la nature des opérations de la requête et la structure des données. La prise en compte de ces paramètres dynamiques et disparates des sites aboutit à un modèle de coût dynamique pour l'optimisation de la requête. Ayant pour objectif de garantir un temps de réponse borné pour la requête, notre approche utilise les sites non-surchargés qui disposent de fragments de la requête. En plus, elle supporte d'une part, la possibilité de répliquer les données de la requête qui se trouvent sur des sites surchargés et d'autre part, l'adaptation du nombre de sites qui interviennent dans le traitement pour réduire le risque de blocage dû à la surcharge d'un site. Notre stratégie de calcul de plan d'exécution d'une requête détermine les sites qui doivent traiter les fragments de la requête et le coordonnateur qui effectuera le traitement global (ou final). Elle a pour objectif de respecter la contrainte de temps de réponse borné lorsque cela est possible en évitant de surcharger les participants et minimise le nombre de participants au traitement de la requête. Pour cela, nous déterminons, dans un premier temps, les sites qui peuvent traiter tous les fragments de la requête sans réplication de donnée avec le respect de la contrainte de temps de réponse. Lorsqu'il n'existe pas de plan qui puisse garantir la contrainte de temps de réponse borné

sans la moindre réplication de fragments, nous cherchons dans un deuxième temps un plan qui permet de garantir cette contrainte en répliquant les fragments qui se trouvent sur des sites surchargés vers d'autres sites. Il importe de noter qu'il est possible que la contrainte de temps de réponse borné ne puisse pas être respectée lorsque les sites sont surchargés et/ou que les temps de transferts sont élevés. Dans ce cas, nous répliquons les fragments qui se trouvent sur des sites surchargés vers le site qui a le plus de données de la requête afin de regrouper davantage les fragments invoquées ensemble tout en augmentant les possibilités de choix pour les requêtes ultérieures. Afin de réduire la quantité de résultats partiels à transférer, notre approche choisit comme coordonnateur de la requête, le site à qui, on a assigné le plus de (sous) traitement de la requête. Ce coordonnateur exécute le traitement global de la requête dès qu'il reçoit tous les résultats partiels distants de la requête et/ou finit ses traitements locaux de la requête.

IV. CAPACITÉ DE STOCKAGE LIMITÉE : UNE STRATÉGIE DE REMPLACEMENT ADAPTÉE

La réplication dynamique des données engendre un surcoût de stockage croissant. Or, la capacité de stockage de chaque site est limitée. Ainsi, nous adoptons une stratégie de remplacement qui adapte le contenu de la base en fonction de sa capacité et des demandes. Cette stratégie conserve seulement dans la base de données les fragments les plus populaires en supprimant ou déplaçant les fragments les moins populaires qui ne sont pas utilisés par un traitement (encours ou en attente), lorsque le contenu dépasse la taille autorisée. La popularité d'un fragment correspond au nombre de fois qu'il a été demandé dans un intervalle de temps donné. Un fragment impopulaire est supprimé si d'autres bases voisines en disposent. Dans le cas contraire, il est déplacé vers un site. La stratégie de remplacement a pour objectif d'optimiser les ressources de stockage tout en garantissant la disponibilité des données sollicitées par les requêtes. Il importe de noter aussi que cette stratégie de remplacement permet d'adapter le degré de réplication de chaque fragment en fonction de son taux d'utilisation. En effet, les fragments les plus sollicités ne seront pas supprimés et seront répliqués lorsque les sites qui les hébergent sont surchargés. Alors que les degrés de réplication des fragments les moins sollicités sont davantage réduits lorsque les sites les abritent sont remplis avec des fragments sollicités.

V. VALIDATION EXPÉRIMENTALE

Nous évaluons la faisabilité et l'adaptation dynamique du schéma de placement dynamique de notre approche en fonction de la popularité des données et des charges des sites.

A. Méthodes expérimentales

1) *Environnement et outils*: Nous avons implémenté notre solution sur une infrastructure composée de onze machines. Chaque machine dispose des caractéristiques suivantes : 22 CPU de 2.394 GHz de fréquence, une mémoire de 62 Go avec comme système d'exploitation Debian 3.2.54-2 x 64. Une

machine qui dispose de toutes les données dans sa base locale représente le portail GBIF. Une deuxième machine génère des requêtes et simule les utilisateurs qui envoient des requêtes. Une troisième machine contrôle les accès à la base de données du GBIF. Les machines restantesinstancient chacune un site du système décentralisé avec ses différents modules. Comme SGBD, nous avons utilisé MonetDB [10]. MonetDB est un SGBD qui compresse et stocke les données en mémoire et offre de bonnes performances en évitant les accès disques souvent coûteux [10].

2) *Données*: Nous avons utilisé un miroir des données géo-référencées du GBIF que nous avons dans une base MonetDB qui représente le portail du GBIF. A partir des données d'occurrences de la base, nous avons construit le catalogue taxonomique pour reprendre la structure hiérarchique des données de biodiversité sur la dimension taxinomique. Pour ce qui concerne l'organisation hiérarchique selon la dimension géo-spatiale, nous avons divisé le globe en neuf cellules représentant chacune un continent. Chaque continent est découpé en neuf cellules qui correspondent aux pays. Enfin, chaque pays est divisé en 9 régions (cellules). Chaque fragment est défini par une espèce et une région. Nous avons considéré les fragments avec plus de 10000 occurrences dans la même région pour avoir des résultats plus significatifs.

3) *Requêtes*: Pour toutes les expériences, nous utilisons le modèle de requête du calcul de la co-occurrence de deux espèces dans une région. Pour cela, une requête implique deux fragments (définis par les prédicats taxinomique (nom de l'espèce) et géographique (numéro de la région)) de la même région. Notre générateur de requête SQL produit au début de chaque expérience un nombre fixe de requêtes qui sont par la suite soumises au système. Pour chaque requête nous mesurons le temps de réponse, le temps de chaque OL, le temps du OG et le temps de chaque éventuelle réplication.

B. Faisabilité : Réduction progressive des temps d'accès au portail GBIF

Pour valider la faisabilité de notre solution, nous évaluons l'évolution des durées totales d'accès aux portail du GBIF par intervalle de temps au cours de l'expérience. Pour cela, nous récapitulons pour chaque requête, la durée totale des transferts (ou réplication) de fragments issus du GBIF vers les sites locaux pendant chaque intervalle de temps de dix minutes au cours de l'expérience. Les résultats 1 nous permet de constater un temps élevé pour les accès à la base au GBIF dans les premiers instants de l'expérience. Ce qui est tout à fait normal, d'autant plus qu'au démarrage aucune donnée n'est disponible dans une base locale. D'où la nécessité d'accéder au GBIF pour répliquer à la demande les fragments sollicités dans les bases locales. Ces données répliquées du GBIF sont utilisées pour les requêtes suivantes qui les invoquent. C'est ce qui explique que les temps d'accès à la base du GBIF diminuent progressivement jusqu'à s'annuler vers la fin de l'expérience. Cette réduction progressive des temps d'accès au GBIF pour les requêtes, prouve l'efficacité de notre approche de répartition dynamique à la demande des données

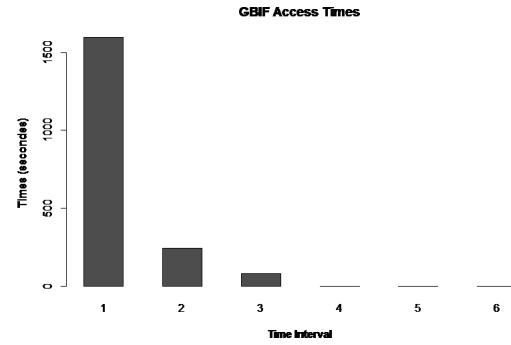


FIGURE 1. Evolution des temps d'accès au portail GBIF par période

en fonction des requêtes. Elle répartit la migration des données dans la période d'exploitation du système et réduit la quantité de données à placer sur les sites locaux lorsque les requêtes ne concernent pas toutes les données de la base du GBIF. Ce qui permet de sauvegarder non-seulement la durée globale de la migration mais aussi de l'espace de stockage à travers le système décentralisé.

C. Adaptation dynamique du schéma de réplication des données

Nous étudions l'adaptation du schéma de placement des données en évaluant la variation du degré de réplication des fragments en fonction des nombres d'accès pendant une expérience. Pour cela, nous incrémentons pour chaque fragment son nombre d'accès à chaque fois qu'il est impliqué dans une requête. De la même façon, nous incrémentons le degré de réplication d'un fragment à chaque fois qu'une nouvelle réplique du même fragment est créée. Après avoir trié les fragments par leurs nombres d'accès, nous représentons le nombre d'accès (figure 2) et le degré de réplication (figure 3) de chaque fragment, puis nous les regroupons par classe (de 15 fragments) en fonction des nombres d'accès et représentons le degré de réplication moyen (figure 4) de chaque classe.

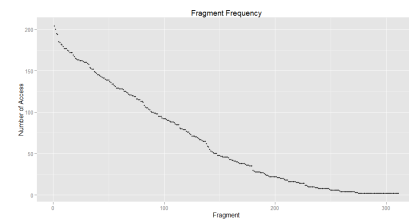


FIGURE 2. Popularité des fragments

Ces figures(2, 3 et 4) montrent que le degré de réplication de chaque fragment (ou classe de fragments) évolue généralement avec la popularité du fragment (ou de la classe). Ce qui prouve que notre mécanisme de distribution adapte le degré de réplication de chaque fragment en fonction de la quantité de requêtes qui l'invoquent. Par ailleurs, on voit que des

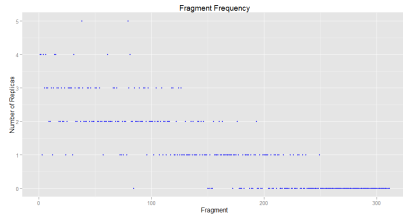


FIGURE 3. Degrés de réplication des fragments

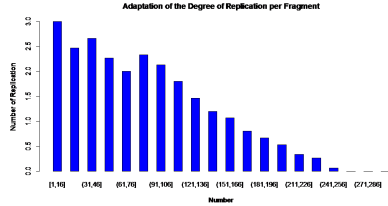


FIGURE 4. Degré de réplication par classe d'accès

fragments plus sollicités sont légèrement moins répliqués que des fragments moins populaires. Ceci s'explique par le fait que la décision de répliquer un fragment dépend aussi de la surcharge des sites qui hébergent ses répliques.

D. Adaptation dynamique de la charge

Basée sur une contrainte de temps de réponse borné, notre solution a pour objectif d'éviter qu'un site soit surchargé. Ainsi, nous avons évalué l'évolution de la charge totale des traitements de chaque site par intervalle de temps de 10 minutes. Cette charge correspond à la somme des durées de traitements assignés au site et des temps de création d'une nouvelle réplique. Les résultats montrent qu'un site n'est pas

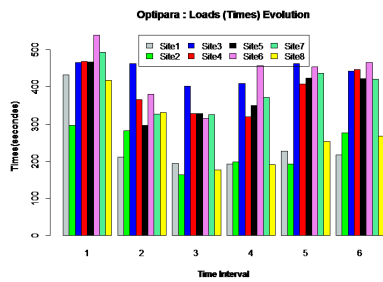


FIGURE 5. Evolution de la charge à travers les sites

surchargé pendant tout un intervalle de temps. De cette façon, notre approche évite de surcharger les sites. La répartition de la charge entre les sites dépend donc de la capacité de chaque site. Cette stratégie ne garantit pas une charge équitable entre les sites, mais elle évite qu'un site soit surchargé en attribuant les nouveaux traitements aux sites libres ou moins chargés qui pourraient garantir un temps de réponse borné.

VI. CONCLUSIONS ET PERSPECTIVES

Dans cet article, nous avons proposé une approche de répartition dynamique des données de biodiversité à la demande dans une architecture répartie à large échelle. Elle exploite la structure hiérarchique des dimensions taxinomique et géographique des données de biodiversité et les spécificités des requêtes d'analyse pour fragmenter les données et est guidée par le processus d'optimisation de la requête pour adapter dynamiquement le placement et la réplication des données en fonction des accès, de la localité et des charges. Nous avons mise en œuvre une stratégie de remplacement pour garantir la disponibilité des données sollicités par les requêtes. Nous avons effectué la validation expérimentale de notre approche de répartition dynamique des données avec des données réelles issues de la base complète du GBIF sur un cluster composé d'une dizaine de machines (200 cœurs). Les résultats de nos expériences prouvent la faisabilité de notre solution et que notre stratégie de répartition dynamique des données à la demande réduit progressivement les accès à la base du portail. Elle adapte aussi le schéma de placement de chaque fragment en fonction de sa popularité et la charge des sites qui contiennent des répliques de celui-ci. Par la suite, il serait souhaitable d'évaluer les performances de notre solution sur un cluster très étendu pour s'assurer de la garantie du passage à l'échelle.

RÉFÉRENCES

- [1] D. Agrawal, A. El Abbadi, S. Das, and A. J. Elmore, "Database Scalability, Elasticity, and Autonomy in the Cloud," in *Proceedings of the 16th International Conference on Database Systems for Advanced Applications DASFAA'11*, 2011.
- [2] C. Gope, "Dynamic data allocation methods in distributed database system," *American Academic Scholarly Research Journal*, vol. 4, no. 6, 2012.
- [3] J. Zhou, N. Bruno, and W. Lin, "Advanced partitioning techniques for massively distributed computation," in *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, 2012, pp. 13–24.
- [4] R. Nehme and N. Bruno, "Automated partitioning design in parallel database systems," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, 2011, pp. 1137–1148.
- [5] C. Farah, Habib and I. Samiul, "Data migration : Connecting databases in the cloud," *ICCIT*, 2012.
- [6] Q. Ke, V. Prabhakaran, Y. Xie, Y. Yu, J. Wu, and J. Yang, "Optimizing data partitioning for data-parallel computing," in *Hot Topics in Operating Systems (HotOS XIII)*. USENIX, 2011.
- [7] N. Bame, H. Naacke, I. Sarr, and S. Ndiaye, "Architecture répartie à large échelle pour le traitement parallèle de requête de biodiversité," in *African Conf. on Research in Computer Science and Applied Mathematics (CARI)*, 2012, pp. 143–150.
- [8] —, "Algorithmes de traitement de requêtes de biodiversité dans un environnement distribué," *Revue africaine de la recherche en informatique et mathématiques appliquées (ARIMA)*, vol. 18, pp. 1–18, 2014.
- [9] —, "Bigbio : Utiliser les techniques de gestion du big data pour les données de la biodiversité," in *African Conf. on Research in Computer Science and Applied Mathematics (CARI)*, 2014, pp. 273–284.
- [10] S. Idreos, F. Groffen, N. Nes, S. Manegold, S. Mullender, and M. Kersten, "Monetdb : Two decades of research in column-oriented database architectures," *IEEE Data Eng. Bull.*, 2012.