

ANALISIS SUBQUERY PROCESSING PADA GRID DATABASE DENGAN SKEMA VO (VIRTUAL ORGANIZATION)

Mitha Budianti¹, Toto Suharto², Ely Rosely³

¹Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Seiring dengan perkembangan kebutuhan bisnis dan pengelolaan data yang kian membesar, diperlukanlah suatu teknik manajemen basis data yang handal dan harus dapat beroperasi secara efektif serta efisien di segala kondisi. Berangkat dari sinilah muncul ide pembuatan grid computing, yang selanjutnya memicu perkembangan di ranah basis data dengan terciptanya teknik grid database

Berbeda dari teknik distributed umumnya dalam hal fleksibilitas dari kolaborasi resource, grid memungkinkan untuk menambah dan mengurangi sumberdaya komputasi maupun penyimpanan dari sistem tanpa mengakibatkan perubahan konfigurasi pada sistem secara keseluruhan.

Berangkat dari kondisi yang dinamis tersebut, dari sisi database, beberapa parameter data dapat berubah selama prosesnya dan mungkin menjadi tidak akurat. Untuk itulah, dibutuhkan suatu skema pemrosesan query-subquery yang dapat beradaptasi dengan lingkungan grid tersebut. Pada penelitian tugas akhir ini, dibahas seputar salah satu skema pemrosesan yaitu VO (Virtual Organization). Pemrosesan dalam VO berlangsung dengan mendekomposisi query (query global) menjadi query-query lokal (yang disebut sebagai subquery) menurut kebutuhan informasi berdasarkan proses bisnisnya, lalu mengalokasikannya ke node-node penyimpanan virtual VO yang sesuai. Sehingga, di sini diuji bagaimana VO dengan skema pemrosesan yang demikian, dapat diimplementasikan untuk lingkungan grid database yang dinamis seperti yang telah disampaikan sebelumnya. Sementara parameter uji yang digunakan yakni dari segi penjaminan availability data termasuk reliability-nya serta toleransi terhadap kinerja waktu eksekusi query-subquery untuk setiap skenario kondisi failure node grid.

Kata Kunci : grid database, grid, vo, virtual organization, subquery, virtualization

Abstract

Along with the growing of bussiness and data management needs, require a reliable database management technique that could be implemented in effective and efficient way for all environment condition. Hence, came an idea of making grid database which inspired by the concept of grid computing.

Distinguished from conventional distributed technique by its focus on flexibility of large-scale resource sharing, grid enables server and storage resources to be added or removed from the system without requiring compicated configuration changes. In this such dynamic environment, some database parameters which are needed in query-subquery processing may be inaccurate and changed during the process in grid database. So we need a query-subquery processing schema that could be adaptive to the grid database environment. For this reasons, the paper brings forward a subquery processing model caled VO (virtual organization). Processing in VO schema decomposed query (here after just called global query) into local query (called subquery) and assign the certain subquery to certain node to be finished. Finally, the paper tests the schema through experiment to see if it is match for the dynamic grid database environment. Herein, the paper uses availability data, reliability, and query-subquery execution time parameter for analyzing VO"s tolerancy in each grid node failure scenario.

Keywords : grid database, grid, vo, virtual organization, subquery, virtualization.

1 PENDAHULUAN

1.1 Latar Belakang

Saat ini, besarnya lingkup komputasi dan data dalam suatu organisasi memicu timbulnya penumpukan sumber daya (*islands of computing*). Penumpukan sumber daya ini justru membawa penurunan efisiensi penggunaan sumber daya – sumber daya tersebut [13]. Sumber daya tersebut terikat secara fisik sementara penggunaannya terkadang justru terpusat ke satu komponen / *site* saja. Sehingga akan sangat sulit dan merepotkan dalam pengelolaannya, terutama saat terjadi *traffic* (penumpukan beban kerja) atau ada komponen / *site* yang berhenti beroperasi. Salah satu solusi yang diajukan untuk menangani isu ini ialah dengan *grid computing* yang di dalamnya juga mencakup *grid database*.

Grid database merupakan suatu bidang penelitian baru yang mengimplementasikan teknik komputasi grid dengan *database*. Dengan teknik komputasi grid, modularisasi perangkat keras [14] memungkinkan untuk skalabilitas, *information sharing*, dan alokasi sumber daya secara lebih luas serta fleksibilitas penanganan *failure* (*failover*) [6].

Node-node fisik dalam grid sendiri antara lain memiliki karakteristik terdistribusi, dan dinamis [5]. Dengan sifatnya yang dinamis, maka dalam pemrosesan *query* tidak dapat menggunakan teknik *query processing* tradisional, melainkan harus dapat menyesuaikan dengan kebutuhan-kebutuhan *query processing grid database* yang dinamis pula [2].

Lebih lanjut, karena tiap node memiliki karakteristik yang berbeda satu sama lain [13], maka biaya yang dibutuhkan tiap node untuk menyelesaikan suatu *query* akan berbeda. Sehingga, salah satu poin utama dalam *grid database* terletak pada bagaimana menentukan pemberian *query* ke node yang tepat [8] (baik dari segi waktu pengaksesan, ketersediaan dan akurasi data sekalipun saat terjadi *failure*).

Salah satu skema pemrosesan query yang ada yaitu VO (*Virtual organizations*). VO merupakan sebuah *global metadata view*, yakni sebuah skema metadata yang diekstrak dari atribut-atribut dalam *database* di node-node grid. Sehingga *query* yang masuk ke dalam grid akan didekomposisi menjadi *query* lokal (atau disini disebut sebagai *subquery*) berdasarkan global metadata tersebut [10].

Pertanyaannya adalah apakah VO ini mampu menjawab tantangan dalam *grid database* itu sendiri yakni dalam hal *failover* dan *availability* saat *downtime* sekalipun. Seberapa cepat skema ini dalam menyelesaikan *subquery* yang masuk ke node-node grid? Apakah skema ini masih mampu menyelesaikan suatu *subquery* ketika terjadi node mati? Jika ya, apakah akurasi datanya masih terjaga?

Hal inilah yang selanjutnya akan menjadi kajian penulis dalam tugas akhir ini. Harapannya VO dapat menjadi solusi untuk diimplementasikan dalam *subquery processing* di *grid database*, mengingat *subquery* sendiri merupakan salah satu teknik penting dalam pemrosesan data khususnya dalam *grid database* yang pada umumnya menggunakan *query* kompleks.

1.2 Perumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah yang menjadi fokus penulis di sini adalah:

1. Apakah VO dapat menangani masalah *failover node* dengan tetap menjaga *availability data* (termasuk akurasi) dalam pemrosesan *subquery* di *grid database*?
2. Seberapa toleran VO terhadap *failure node* termasuk bagaimana kinerjanya di *grid database* pada kondisi tersebut?

1.3 Tujuan

Tujuan yang ingin dicapai dalam tugas akhir ini, meliputi :

1. Skema VO dapat diimplementasikan pada *grid database*.
2. VO menjadi solusi pemrosesan subquery di *grid database* yang dapat menjaga data *availability data* untuk setiap node yang tergabung di grid infrastruktur sehingga layanan akan selalu tersedia bagi semua unsur yang berkolaborasi di dalamnya.

1.4 Batasan Masalah

Dalam implementasi tugas akhir ini dibatasi oleh beberapa hal, antara lain :

1. Skema pemrosesan query dalam *grid database* menggunakan skema VO dengan *vertical partitioning*.
2. Studi kasus yang digunakan yakni seputar proses bisnis pada *Endangered Species Conservation (ESC)*.
3. Data yang digunakan merupakan data *dummy*, hasil *generate* otomatis dengan menggunakan tool pembantu, Data Generator.
4. Jenis query yang dipakai yakni query *selection* dengan subquery *centralized node subquery* dan *distributed node subquery*.
5. Analisis yang dilakukan yakni terhadap kinerja waktu pemrosesan (*response time*), *availability* (termasuk akurasi) pada kondisi normal (tanpa *failure*) dan terjadi *failure*.
6. Uji *failure* bermain di sisi *grid storage* (ketersediaan *resource* di sisi grid storage masih lebih banyak dibanding di sisi grid komputasi).
7. Skenario test *availability* dilakukan dengan meng-online offline-kan node-node *virtual storage* (ASM Disk) yang diasumsikan sebagai *physical disk storage*.
8. DG (*Disk Group*) merupakan sekumpulan ASM Disk (*AD*), dimana 1 AD = 1 LUN = 1 VD (*Volume Disk*).

1.5 Metodologi Penyelesaian Masalah

Penelitian ini akan menunjukkan bahwa VO dapat diterapkan untuk pemrosesan *subquery* pada *grid database*. Dengan parameter kajian dari sisi *availability* data dan waktu pemrosesan *subquery*.

Adapun alur metodologi penyelesaian masalah yang akan digunakan yaitu :

- a. Studi Literatur :
Mempelajari literatur-literatur yang ada sesuai dengan permasalahan meliputi konsep dari *subquery processing*, *Virtual organization (VO)*, dan *grid database*.
- b. Perancangan sistem :
Membuat desain skema *environment grid database* yang akan dibangun, serta skenario pengujian.
- c. Implementasi :
Membangun infrastruktur *grid environment* berikut *database grid* dengan DBMS Oracle.
- d. Testing dan analisis hasil, meliputi :
Melakukan pengujian kinerja terhadap sistem yang telah dibangun menurut skenario yang dirancang sebelumnya.
- e. Analisa hasil :
Analisa terhadap hasil pengujian yang didapat, dengan melakukan analisa dari segi waktu pemrosesan (*execution time*) serta ketersediaan data saat node mati (*failover node*) untuk setiap skenario pengujian.
- f. Pembuatan laporan :
Pengambilan kesimpulan dari hasil analisa yang didapat untuk selanjutnya dituangkan dalam laporan hasil pengerjaan Tugas Akhir.

1.6 Sistematika Penulisan

Tugas akhir ini disusun dengan sistematika penulisan sebagai berikut:

- BAB I PENDAHULUAN**
Berisi pemaparan mengenai latar belakang permasalahan, tujuan yang ingin dicapai melalui tugas akhir kali ini, perumusan masalah, batasan masalah, metodologi tugas akhir, dan sistematika penulisan.
- BAB II LANDASAN TEORI**
Berisi uraian seputar landasan teori yang digunakan, meliputi teori tentang *Grid database*, Skema *Virtual organization* dan teori-teori lain yang berkaitan dengan penelitian tugas akhir ini.
- BAB III ANALISIS PERANCANGAN SISTEM**
Berisi perancangan sistem dan infrastruktur *Grid database Environment* yang akan dibangun serta skenario pengujian yang dilakukan.
- BAB IV ANALISIS IMPLEMENTASI DAN PENGUJIAN SISTEM**
Berisi implementasi dari skenario yang telah dibuat, termasuk analisa hasil pengujiannya.
- BAB V PENUTUP**
Berisi kesimpulan dan saran-saran untuk pengembangan lebih lanjut terhadap hasil penelitian tugas akhir ini.

5 KESIMPULAN DAN SARAN

5.1 Kesimpulan

Menurut hasil pengujian untuk *Subquery Processing* pada *Grid database* dengan Skema *Virtual organization* (VO) pada tugas akhir ini, dapat ditarik kesimpulan sebagai berikut :

1. Skema VO dapat diimplementasikan untuk pemrosesan *subquery* pada *grid database* dengan studi kasus kali ini mengenai *Endangered Species Conservation*.
2. Dengan skema VO, *availability* termasuk akurasi data pada *grid database* untuk kondisi *failure* sekalipun dapat terpenuhi. Baik untuk pemrosesan *subquery* jenis *distributed node subquery* maupun *centralized node subquery*.
3. Untuk kinerja skema VO dalam melakukan *subquery processing* juga dipengaruhi oleh jumlah node *offline* yang dapat ditolerir. Makin banyak node yang *offline* pemrosesannya akan menjadi makin lambat.

5.2 Saran

Terakhir, berangkat dari proses dan hasil yang didapatkan dalam penelitian kali ini, ada beberapa saran yang dapat penulis sampaikan untuk perbaikan penelitian di bidang serupa, antara lain :

1. Pengujian *availability* untuk level *database*. Maksudnya, skenario yang dapat diatur disini yaitu dengan mencoba mematikan satu atau beberapa *database* yang berkolaborasi dalam *grid database environment*. Tentunya untuk melakukan ini dibutuhkan *resource* yang lebih banyak dan besar.
2. Parameter yang diuji dapat lebih dieksplorasi. Bagaimanakah skema VO ini bila dilihat dari parameter lain semisal dari segi *security*, *concurrency* data, ataupun optimasi *query-subquery* yang digunakan.

6 Daftar Pustaka

- [1] Best, T., dan M.J. Billings. 2005. *Oracle Database 10g : Administration Workshop I*. USA : Oracle.
- [2] Bougamin, L., Francoise Fabret, C. Mohan, dan Patrick Valduriez. 2000. *A Dynamic Query Processing Architecture for Data Integration Systems*. France : Computer Society, IEEE
- [3] Boyd, D., dan Alistair Mills. 2003. *Building an e-Science Grid for the UK*. UK : Grid Engineering Task Force.
- [4] Chan, Immanuel. 2006. *Oracle Database High Availability Overview*. USA : Oracle
- [5] Czajkowski, K., Steven Fitzgerald, Ian Foster, dan Carl Kesselman. 2001. *Grid Information Services for Distributed Resource Sharing*. USA: ComputerSociety, IEEE
- [6] Foster, I., Carl Kesselman, dan Steven Tuecke. 2001. *The Anatomy of the Grid, Enabling Scalable Virtual Organization*. USA : The University of Chicago
- [7] Gehrke, J., dan Raghuram Ramakrishnan. *Database Management System Second Edition*. New York : McGraw Hill.
- [8] Gournis, A., R. Sakellariou, N. W. Paton, dan A. A. A. Fernandes. 2004. *Resource Scheduling for Parallel Query Processing on Computational Grids*. UK: Computer Society, IEEE
- [9] Goyal, Brajesh. 2003. *Oracle Database 10g : The Database for the Grid*. USA : Oracle Corporation
- [10] Hu, N., Yingying Wang, dan Liang Zhao. 2007. *Dynamic Optimization of Subquery Processing in the Grid Database*. China : Third International Conference on Natural Computation.
- [11] Microsoft Encarta Reference Library. 2007. *Reliability and Data*. <http://www.barringer1.com/jul07prb.htm>. Diakses pada tanggal 4 maret 2011.
- [12] Nash, Miranda. 2003. *Oracle 10g: Infrastructure for Grid computing*. USA: Oracle Corporation.
- [13] *Partitioning Enhancements in Oracle Database 11g Release 1*. http://www.oracle-base.com/articles/11g/PartitioningEnhancements_11gR1.php. Diakses pada 6 Mei 2011.
- [14] Paton, N. W., A. Gounaris, A. A. A. Fernandes, dan Rivos S. 2002. *Adaptive Query Processing: A Survey*. UK : SpringerLink.
- [15] Rahayu, Nur Wijayaning. 2005. *Arsitektur Grid computing pada Oracle 10g*. Yogyakarta : Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005).
- [16] Saptono, Henry. *Openfile distro khusus SAN/NAS (bagian 2)*. Didownload pada 26 Januari 2011
- [17] Silberschatz, Korth, Sudarshan. 2001. *Database System Concepts, Fourth Edition*. New York : McGraw-Hill.

- [18] Taniar, D., Clement H.C. Leung, Wenny Rahayu, dan Sushant Goel. 2008. *High-Performance Parallel Database Processing and Grid Databases*. USA : Wiley.
- [19] *Testing for Availability*. [http://msdn.microsoft.com/en-us/library/aa292184\(v=vs.71\).aspx](http://msdn.microsoft.com/en-us/library/aa292184(v=vs.71).aspx). Diakses pada tanggal 9 Maret 2011.
- [20] Thorpe, Matt. 2005. *Engineering Task Force*. <http://tyne.dl.ac.uk/ETF/index.shtml>. Didownload tanggal 23 Februari 2011.
- [21] VMWare ESX and VMWare ESXi. 2001. *The Market Leading Production-Proven Hypervisor*. USA : VMWare Inc.
- [22] Wikipedia. 2009. *Grid computing*. http://en.wikipedia.org/wiki/Grid_computing. Diakses pada 21 November 2010.
- [23] Yu, K., dan David Mar. 2009. *Building Oracle Grid with Oracle VM on Dell Blade Server and Dell EqualLogic iSCSI Storage*. USA : Dell Database Solutions Engineering.
- [24] Zitelli, A. V. 2010. *Oracle 11g Reference Partitioning - Benefits, Hazards & Other Considerations*. California : Thales Raytheon System.