

Expert System pada Komputer paralel

Jani F Mandala¹⁾

¹⁾Jurusan Teknik Elektro, Fakultas Sains dan Teknik, Universitas Nusa Cendana
Jl. Adisucipto, Penfui-Kupang NTT, Indonesia
Email: janifm99@yahoo.co.id

Abstrak

Pemrosesan data berskala besar dan berbeda-beda sistem sangatlah diperlukan, untuk itu pemilihan proses komputer secara paralel sangat diperlukan. Sebagai acuan pada penulisan ini didekati dengan penggunaan expert system sebagai bagian terkecil dalam sistem relasional data yang melakukan pemroses keputusan. Proses ini digunakan sistem komputer paralel dimana data-data sebelum diproses, diparsialkan dengan algoritma paralel kemudian memanfaatkan beowulf dan pemrogram PVM (*paralel virtual mechine*) untuk proses komputer paralel. Melalui uraian ini, diperoleh expert system dapat dikembangkan pada sistem komputer paralel menjadi multi platform dan lebih fleksibel dengan membentuk sejumlah paket berukuran matriks. Pemrosesan dapat mengetahui pemrograman paralel dan durasi proses data pada jaringan dan penggunaan komputer paralel tersirat reduksi penggunaan sistem yang baru.

Kata Kunci : Relasional data, sistem pakar, beowulf dan PVM

Abstract

Large-scale data processing and the different systems is necessary, for the electoral process in parallel computer is indispensable. As a reference in this paper was approached by the use of expert systems as part of the smallest in the relational data systems perform decision processing. This process is used parallel computer system in which the data before it is processed, parallel algorithm then utilizes Beowulf and PVM (paralel virtual mechine) programmers to process parallel computer. Through this description, obtained expert systems can be developed on a parallel computer system into a multi-platform and more flexible by forming a matrix-sized package. Processing can determine the duration of the process of parallel programming and data on the network and the use of parallel computers drained reduction of the use of the new system

Keywords: Relational of data, expert systems, Beowulf and PVM

PENDAHULUAN

Interkoneksi antara PC (*personal computer*) sebagai olahan data, saat ini sangat diperlukan untuk merespon data-data sebagai simpulan. Walaupun data-data tersebut berbeda-beda dan dalam ukuran yang berskala besar.

Permasalahan yang dimaksudkan disini adalah pemrosesan basis data yang dapat mengevaluasi keadaan (kondisi), Untuk menaggulangi permasalahan ini, digunakan pendekatan secara pustaka antara *expert system* (sistem pakar) dengan relasional data dan komputer paralel.

Kepakaran yang tercakup disini merupakan rangkuman hasil interpretasi PC terhadap pengetahuan sebagai saran-saran yang dapat digunakan. Dan dalam prosesnya, bagian pada *inference engine* yang ditekankan sebagai

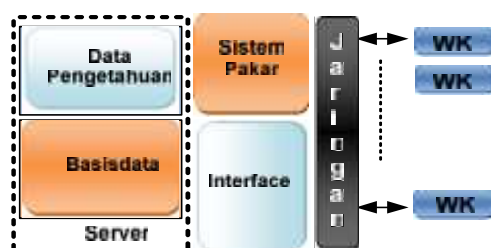
pengujian setiap PC tanpa perlu dikondisikan dalam aturan yang rumit tetapi sebagai proses yang didasari oleh adanya permintaan relasional data yang kompitibel dengan bahasa *query*. Sedangkan pada komputer paralel, merupakan proses olahan data yang dilakukan secara serempak oleh beberapa PC.

SISTEM PAKAR

Expert system (sistem pakar) disini, bertindak sebagai rangkuman pengetahuan yang terdapat pada penyimpanan. Solusi dari sistem ini, melalui bebarapa tahapan, diantaranya basis pengetahuan (*knowledge base*) dan *inference engine*.

Basis pengetahuan pada sistem pakar, merupakan fakta-fakta yang merepresentasikan jawaban yang logis melalui kaidah kondisi

(statement) atau *if then*. Sedangkan pada *inference engine* merupakan simpulan pencocokan data secara siklus maju maupun mundur yang diturunkan dari basis pengetahuan. *Inference engine* dilengkapi dengan relasional data dan basis pengetahuan itu sendiri yang merupakan kelompok data terstruktur (hirarkir) dengan proses *rule-rule* logik. Jadi secara tidak langsung disini, proses relasional bisa meniadakan hirarkir data dan prosesnya berupa relasi tabel-tabel.



Gambar 1. Diagram Proses Pakar-Paralel

Proses *rule-rule* logik ini memiliki keterkaitan dengan *rule-rule* logik yang lainnya, jadi jika salah satu sudah mendapatkan solusi tambahan maka aturan yang lainnya akan dimodifikasi oleh *inference engine*. Proses ini berlanjut sehingga membentuk siklus dengan solusi yang baru, sehingga ragam fasilitas diperlukan untuk proses manipulasi variabel-variabel (Gambar 1) dengan ruang dan olahan proses tersendiri tanpa melakukan perubahan substansi data. Serta data-datanya merupakan pengembangan dari relasional data melalui pendeklarasian dan penelusuran.

Dengan rumusan ini, maka proses pada *inference engine* dapat dikembangkan dengan macro yang ada pada relasional *database* sebagai proses dengan data eksternal.

KOMPUTER PARALEL

Komputer paralel merupakan proses manipulasi secara bersama-sama atau serempak (paralel). Proses ini dilakukan dengan membagi-bagi program menjadi beberapa bagian dan diproses (beban kerja atau task) oleh beberapa prosesor (N) dengan N banyaknya proses. Dengan cara ini, maka *output* dapat diperoleh sesegera mungkin.

$$S_n = \frac{N}{1+(N-1)\alpha}$$

Dimana ;

S_n = peningkatan kecepatan

N = jumlah prosesor

α = fraction of bootlencek

Umumnya komputer paralel dapat dibagi menjadi beberapa kelompok, diantaranya; multi komputer, multi prosesor dan, proses array. Pada kasus ini, hanya ditekankan pada kelompok multi-komputer atau proses yang terjadi pada beberapa komputer dengan penekanan beban kerja pada prosesor dan memori itu sendiri.

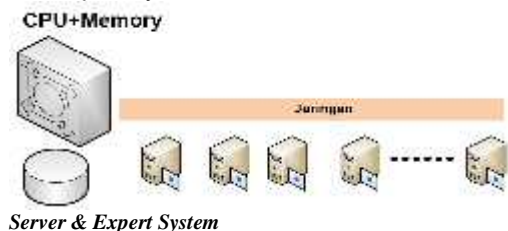
Proses kerja pada paralel paling tidak adanya kesesuaian arsitektur dan algoritma. Menurut beberapa pandangan, langkah awal adalah penentuan algoritma dan langkah berikutnya perancangan komputer paralel. Bagian awal disertai dengan pendefinisian bagian-bagian komputasi, selanjutnya bagian ini diparsialkan sehingga memungkinkan untuk melakukan proses. Langkah berikutnya dilakukan kalkulasi ulang untuk mendapatkan beberapa bagian proses yang perlu dilakukan penggabungan atau tidak. Bila semua telah *real* maka proses selanjutnya diimplementasikan.

Message passing merupakan langkah pembuatan proses (task) untuk komunikasi dan selanjutnya digunakan fungsionalitas *multithread* yang ada pada bahasa pemrograman, sebagai program yang dapat mengeksekusi secara bersamaan. Didalam *message passing* ini tidak terlepas dari *mapping* yang meningkatkan proses dengan cara pemetaan *message passing* sebagai proses secara bersamaan.

Tujuan utama dari hal di atas adalah membagi data menjadi bagian-bagian kecil (parsial) untuk dapat diproses secara bersamaan pada prosesor yang berbeda dan kemungkinan proses operasi ini akan sering dilakukan pada prosesor yang sama.

Pada sisi yang lain dari sistem komputer paralel adalah; implementasi *software* paralel. Sistem ini masih dapat diandalkan untuk melakukan pengujian dengan jumlah node yang sederhana; *beowulf (linux cluster)* dengan kecepatan dan performansi yang tinggi dan standar pemrogramannya *PVM (paralel virtual mechine)*. Pendekatan ini mengingat sama-sama

sistem multi-komputer yang mengerjakan olahan secara bersamaan, walaupun sistem pada *beowulf* hanyalah *server* (Gambar 2).



Gambar 2. Jaringan Multi-Komputer

Pada bagian *PVM* terdiri dari *daemon* dan *library*, *daemon* merupakan layanan sistem sedangkan *library* sebagai fungsi untuk menjalankan tugas-tugas pemrograman.

IMPLEMENTASI SISTEM.

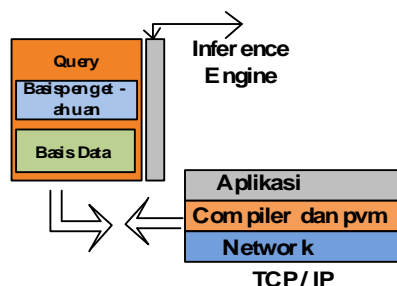
Expert System.

Berdasarkan rujukan di atas, maka interaksi *inference engine* bekerja berdasarkan *server* dan *workstation (user)*. *Server* sebagaimana fungsinya yakni menampung, distribusi (permintaan) dan pemuktahiran data-data. *Server* digunakan *macro* yang ada pada relasional databases (sql), misalnya; *ms server*, *fox pro*, *mysql ver5* dan lain-lain. Kegunaannya untuk membagi tugas-tugas (task) pada masing-masing *workstation* atau *user (client)*. Berdasarkan Gambar 1 dan 2, maka server memiliki dua bagian data terbesar yakni basis pengetahuan sebagai data-data utama (databases) sedangkan *inference engine* yang ter-include dengan bagian terkecil dari data itu sendiri sebagai basis data pengetahuan (pakar). Data-data ini diproses berdasarkan permintaan beban kerja yang diletakan pada *inference engine* yang bertindak sebagai *user interface* (Gambar 3).

Multi Komputer

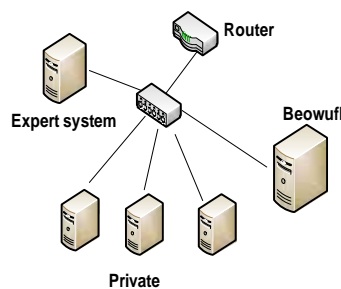
Sistem pada Gambar 2 dan 3, menunjukkan peran *software* dan *hardware* sangat memungkinkan dalam proses konfigurasi. Hal ini lebih menampilkan keseragaman sistem dengan tujuan untuk kemudahan aplikasi (program) dan konfigurasi

jaringan. Mengingat pada bagian tertentu (*user*) akan bertindak sebagai *dumb* (hanya proses).



Gambar 3. Sistem Pakar pada TCP/IP

Jaringan dengan konfigurasi sebagai *bus* atau bintang dapat memanfaatkan *beowulf* yang diatur *ip private* dan publik menggunakan *router*.



Gambar 4. Sistem Jaringan Beowulf

Sistem jaringan *beowulf (cluster)* ter-include dengan *PVM* yang dapat juga menangani proses text dan numerik secara interoperabilitas sedangkan pada sisi yang lain menjamin sinkronisasi *time*.

Uji coba pada sistem (Gambar 4) lebih ditekankan pada pertukaran data melalui *inference engine* yang berjalan juga pada *PVM*. Walaupun proses ini nantinya pada proses, akan digunakan kepadatan jaringan dengan durasi proses pemaketan data *timing* dari awal hingga akhir proses pemaketan. Proses ini juga akan menjadi terukur bila diketahui dari pemrograman yang dihasilkan (fungsional). Mengingat untuk setiap proses dilakukan dekomposisi data dalam ukuran berupa matriks, sehingga proses akan terdapat kesesuaian baris dan kolom. Akhir dari proses ini akan kembali

ke server *expert system* (pembagian *timing*) dengan nilai *big O*.

SIMPULAN

- Berdasarkan konfigurasi pakar ini, dapat dikembangkan basis jaringan dengan mengintegrasikan data eksternal dan dapat dikembangkan menjadi multi platform yang lebih fleksibel.
- Untuk proses secara paralel, sistem pakar dapat dilakukan pada banyak prosesor dengan ukuran matrik sebagai gambaran jumlah pemaketan data yang terjadi.
- Dengan penggunaan komputer paralel tersirat reduksi pembebanan dan penggunaan perangkat yang baru.

DAFTAR PUSTAKA

- [1] Ajinagoro, Bagus Irawan, "Aplikasi Sistem Paralel Menggunakan Prosesor Host 486 Berbasis Linux Debian", Tugas Akhir STT Telkom, Bandung., 2005
- [2] Brown G, Robert, "Engineering a Beowulf-Style Compute Cluster". Duke University Physics Dept.,
- [3] url: www.phy.duke.edu/~rgb/Beowulf/beowulf_book.php., 11 Pebruari 2015.
- [4] Hwang, Kai, "Advanced Computer Architecture Paralism, Scalability, Programmability", McGraw-Hill. 1993.
- [5] Lewis, Ted, Kyu, Kim dkk., "Introduction To Parallel Computing, Printice Hall., 1992.
- [6] Mandala, Jani F., "Simulasi Message Passing Multicomputer". Lingkungan Semirangkai, ISSN 0852-4874., September 2009.
- [7] Rewini., Hesham, Lewis, G. Theodore., Ali, Hesman H., "Task Scheduling In Parallel and Distributed System"., PTR Prentice Hall., New Jersey., 1994.
- [8] Rochman, Abdul., "Paralel Blok Faktorisasi QE Dalam Sistem Memori Terbesar Multi Komputer Berbasis MPI-Linux"., Jurnal Informasitika Vol. 8, No. 2, Nopember 2007: 134 – 138.
- [9] Tamer., M Ozsu., "Principles of Distributed systems"., Paris, France., Prentice Hall., 1992.
- [10] Woodman, Lawrence., "Setting up a Beowulf Cluster Using Open MPI on Linux"., 2 December 2009., Url : <http://techtinkering.com/archive/>., 11 Pebruari 2015