Page 1

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 59 http://www.euroasiapub.org MEMBANDINGKAN MANUAL DAN OTOMATIS NORMALISASI TEKNIK UNTUK DATABASE RELASIONAL Sherry Verma \* Abstraksi Normalisasi adalah proses menganalisis skema relasi yang diketahui berdasarkan Fungsional mereka dependensi dan kunci utama untuk mencapai sifat yang diinginkan untuk meminimalkan redundansi. Hal ini bertujuan untuk menciptakan satu set tabel relasional dengan minimum redundansi data yang melestarikan konsistensi dan memfasilitasi pemasangan yang benar, penghapusan, dan modifikasi. A Database normalisasi tidak menunjukkan berbagai sisipan, penghapusan dan anomali modifikasi karena masa depan pembaruan. Makalah ini menyajikan studi perbandingan manual dan otomatis Teknik normalisasi menggunakan sekuensial serta algoritma paralel. Hal ini sangat banyak waktu memakan waktu untuk menggunakan teknik otomatis untuk melakukan hal ini analisis data, sebagai lawan melakukannya manual. Pada saat yang sama, proses tersebut diuji dapat diandalkan dan benar. Ini menghasilkan matriks ketergantungan dan grafik matriks diarahkan, pertama. Ini kemudian melanjutkan dengan menghasilkan 2NF, 3NF, BCNF dan bentuk normal. Semua tabel juga dihasilkan sebagai prosedur hasil. Kata kunci: Normalisasi Otomatis, manual Normalisasi, Relational Database, Ketergantungan Fungsional, dan Primary Key. \* Dosen, Ansal Institute of Technology, Gurgaon.

Page 2

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 60 http://www.euroasiapub.org 1. PENDAHULUAN Normalisasi database adalah proses transformasi data ke dalam well-formed atau alami pengelompokan seperti yang satu fakta disimpan di satu tempat [1]. Normalisasi umumnya menyederhanakan hubungan dan mengurangi bahaya anomali [2] yang mungkin sebaliknya terjadi selama manipulasi hubungan dalam database relasional. Selain itu, data dinormalisasi stabil dan, oleh karena itu, memberikan dasar yang baik untuk setiap pertumbuhan di masa depan. Dengan demikian, normalisasi Prosedur memberikan desainer database dengan kerangka formal untuk menganalisis hubungan skema berdasarkan kunci mereka dan dependensi fungsional antara atribut mereka [3]. Ini juga menyediakan serangkaian tes bentuk normal yang dapat dilakukan pada hubungan antar pribadi skema sehingga database relasional dapat dinormalisasi untuk setiap tingkat yang diinginkan. EFCodd pertama diformalkan proses normalisasi. Dibutuhkan skema relasi melalui serangkaian tes untuk sertifikasi apakah itu memenuhi bentuk normal tertentu. The formof yang normal relasi merujuk dengan kondisi bentuk normal tertinggi yang memenuhi, dan karenanya menunjukkan sejauh mana itu telah dinormalisasi. Tiga bentuk normal yang disebut pertama (1NF), kedua (2NF), dan ketiga (3NF) bentuk yang normal pada awalnya diusulkan. Sebuah amandemen kemudian ditambahkan ke normal ketiga bentuk oleh R. Boyce dan EF Codd disebut Boyce-Codd Normal Form (BCNF). Kecenderungan mendefinisikan bentuk normal lainnya terus berlangsung sampai bentuk normal kedelapan. Dalam prakteknya, bagaimanapun, database dinormalisasi sampai dengan BCNF. Suatu relasi ada pada bentuk normal pertama jika setiap lapangan hanya berisi nilai-nilai atom, yaitu, tidak ada daftar atau set, dalam arti bahwa hal itu seharusnya tidak dapat harus dibagi menjadi lebih dari satu nilai tunggal. Setiap bentuk normal kecuali 1NF didefinisikan di atas bentuk normal sebelumnya. Artinya, sebuah tabel dikatakan dalam 2NF jika dan hanya jika berada dalam 1NF dan memenuhi kondisi lebih lanjut. Kecuali untuk 1NF, bentuk normal lainnya didasarkan pada Dependensi Fungsional (FD) antara atribut relasi. Sebuah fungsional ketergantungan (FD) merupakan kendala antara dua set atribut dalam relasi dari database [4]. Mengingat R relasi, satu set atribut X di R dikatakan fungsional menentukan lain atribut Y, juga di R, (ditulis X → Y) jika, dan hanya jika, setiap nilai X berhubungan dengan tepat satu nilai Y. X dikatakan set penentu dan Y set tergantung. Mengingat bahwa A, B, dan C adalah set atribut dalam relasi R dapat diturunkan beberapa sifat fungsional dependensi. Di antara yang paling penting adalah aksioma Armstrong. Aksioma ini digunakan dalam normalisasi database: Axiomof Refleksivitas: Jika B adalah himpunan bagian dari A, maka A -> B Axiomof Augmentation: Jika A -> C, maka AB -> CB Axiomof Transitivitas: Jika A -> B dan B -> C, maka A -> C

Page 3

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 61 http://www.euroasiapub.org Dekomposisi Axiomof, atau proyeksi: IfA -> BC, maka A -> B dan A -> C Axiomof Pseudotransitivitas: Jika A> B, CB-> D, maka AC -> D Normalisasi adalah tugas utama dalam desain database relasional [4]. Mekanisme Proses normalisasi menghemat sejumlah besar waktu dan uang. 2. TRADISIONAL DAN OTOMATIS NORMALISASI TEKNIK 2.1 Pendekatan Tradisional Seperti telah disebutkan, kecuali 1 st Sisanya bentuk normal semua bentuk yang normal tergantung pada FD'S. Dalam metode tradisional normalisasi kita mengikuti definisi abstrak ini biasa bentuk dan menerapkannya untuk beton masalah [5]. Untuk menunjukkan hal ini, mempertimbangkan berikut contoh, informasi mengenai Pemasok, bagian, dan pengiriman yang disimpan dalam relasi tunggal: Pertama (S #, Status, City, P #, Quantity) Di sini kita mengasumsikan bahwa relasi dalam 1 st bentuk normal. Sekarang menurut definisi 2 nd bentuk normal, setiap atribut non-prime harus berfungsi penuh tergantung pada primer kunci. Jadi pertama untuk mengidentifikasi atribut atau kombinasi atribut membuat primary key kita perlu menjalankan algoritma [6] berikut: Sebuah relasi R universal dan satu set ofFD'S F pada atribut R .Untuk menemukan kunci K 1 Set K: = R 2 Untuk setiap atribut A di K {Hitung (KA) + wrt F; IF (KA) + berisi semua atribut di R, kemudian mengatur K: = K {A}}; Kami menemukan bahwa kombinasi (S #, P #) membuat primary key dari relasi ini. Kami mengingat hubungan melanggar kondisi 2NF, sebagai kota dan status sebagian tergantung pada primary key, yang akan menyebabkan redundansi serta anomali. Jadi sesuai dengan aturan kami menghapus atribut sebagian tergantung dari hubungan kami asli dengan menguraikan ke dua relasi sebagai berikut: First1 (S #, P #, Quantity) First2 (S #, Kota, Status) Kedua hubungan di atas adalah di 2NF.Now untuk hubungan berada di 3NF, setiap atribut nonprima harus non-transitif tergantung pada primary key [7]. Jadi menurut definisi ini bahaya berlari ke berbagai anomali di First2 masih ada, namun. Sebagai contoh, jika tuple yang memberikan kota tertentu memiliki status khusus yang perlu dimasukkan, itu tidak dapat

Page 4

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 62 http://www.euroasiapub.org dimasukkan sampai beberapa pemasok benar-benar bergerak ke kota (anomali penyisipan). Jika hanya satu pemasok adalah di kota dan pemasok yang dihapus, maka informasi tentang status yang kota hilang (penghapusan anomali). Akhirnya, jika status untuk kota tertentu perlu diubah, setiap tupel untuk kota yang harus terletak dan berubah (update anomali). Alasan untuk bahaya anomali di First2 adalah ketergantungan transitif status pada S # melalui kota. Setiap S # nilai menentukan nilai KOTA, dan bahwa nilai CITY menentukan nilai STATUS. The solusi untuk masalah ini adalah untuk menguraikan hubungan First2 menjadi dua relasi: First21 (S #, City) First22 (Kota, Status) Kedua hubungan di atas adalah dalam 3NF. Selain itu dekomposisi adalah kerugian kurang dan ketergantungan melestarikan. Sayangnya, pendekatan tradisional ini sulit bagi banyak IS / IT untuk siswa pegang dan / atau menerapkan definisi. Mereka tidak bisa membedakan antara tiga bentuk yang normal dan bingung tentang hubungan antara FD dan bentuk normal. Jadi Hsiang-Jui Kung dan Han Reichgelt mengusulkan metode alternatif. 2.2 normalisasi utomatic sebuah pproach menggunakan algoritma sekuensial Seperti ini banyak pendekatan lain yang diperkenalkan. Tapi meskipun normalisasi penting, sangat sedikit algoritma telah dikembangkan untuk digunakan dalam desain komersial Alat normalisasi otomatis. Algoritma normalisasi Matematika diimplementasikan dalam [8]. Dalam [9] satu set stereotip dan menandai nilai-nilai yang digunakan untuk memperpanjang UML meta-mode. Sebuah grafik aturan penulisan ulang ini kemudian diperoleh untuk mentransfer data model dari satu bentuk normal lebih tinggi bentuk normal. Kemudian Amir Hassan Bahmani datang dengan database otomatis Teknik normalisasi, yang menggunakan ketergantungan diagram grafik untuk mewakili fungsional dependensi database [10]. Pendekatan ini menggunakan tiga struktur, Ketergantungan Grafik (DG), Dependency Matrix (DM), dan Sutradara Grafik Matrix (DG), untuk mewakili dan memanipulasi dependensi antara atribut relasi. Setelah menghasilkan DG matriks kami mengalihkan perhatian kita terhadap menemukan Determinan transitif kunci dependensi matriks yang akan tampilkan ke-dependensi transitif antara kunci penentu. 2NF dan 3NF yang diusulkan Proses normalisasi memanfaatkan baik ketergantungan dan transitif kunci penentu dependensi. Untuk menghasilkan bentuk 2NF, kita harus menemukan semua dependensi parsial; itu diasumsikan bahwa tabel sudah dalam bentuk 1NF. Dalam rangka untuk mengubah hubungan menjadi 3NF, setiap DM dipindai baris demi baris mulai dari baris pertama. Jika kunci penentu adalah ditemui yang ketergantungan yang tidak parsial (dari Gambar 16) tidak juga sepenuhnya tergantung pada bagian dari primary key tabel terpisah harus dibentuk. Tentu saja, jika meja adalah sebelumnya membentuk duplikat tidak dihasilkan. Tabel baru ini akan mencakup determinan

Halaman 5

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 63 http://www.euroasiapub.org kunci dan semua atribut lainnya, yang transitif, tergantung pada tombol ini. Untuk hubungan dengan hanya satu kunci kandidat, 3NF dan BCNF adalah setara. Mari kita ambil contoh sebelumnya dan menyelesaikannya dengan menggunakan teknik yang disebutkan di atas: Pertama (S #, Status, City, P #, Quantity) S #, P # -> Quantity Kota -> Status S # -> City, Status Representasi grafis dari dependensi: -------------------------------------------------- ----------------------------------- Fig1: representasi grafis dari dependensi Jika kita bisa mendapatkan semua dependensi antara kunci penentu kita dapat menghasilkan semua ketergantungan antara semua atribut dari suatu relasi. Dependensi ini diwakili oleh menggunakan Matrix Dependency (DM). Menggunakan jalan menemukan algoritma dan Armstrong aturan transitivitas dependensi baru ditemukan dari ketergantungan set yang ada. Ketergantungan Matrix: Dari grafik ketergantungan, Ketergantungan Matrix yang sesuai (DM) yang dihasilkan sebagai berikut: i. Tentukan matriks DM [n] [m], di mana n = jumlah Determinan Keys. m = jumlah Keys Simple. S Status Kota P SP Kuantitas

Halaman 6

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 64 http://www.euroasiapub.org S P CITY STATUS JUMLAH SP 2 2 0 0 1 CITY 0 0 2 1 0 S 2 0 1 1 0 Disutradarai Matrix Graph: The Directed Graph (DG) matriks untuk kunci penentu digunakan untuk mewakili semua kemungkinan langsung dependensi antara kunci penentu. Ditjen adalah n × n matriks dimana n adalah jumlah kunci penentu. Setelah menjalankan algoritma untuk memproduksi grafik DG, kita mendapatkan berikut DG matriks: SP CITY S SP 1 - 1 1 CITY - 1 1 - 1 S - 1 1 1 Setelah menghasilkan matriks DG kita mengalihkan perhatian kita terhadap menemukan semua jalur yang mungkin antara semua pasangan. Matriks ini akan menampilkan semua dependensi transitif antara determinan kunci. Ada banyak jalan menemukan algoritma seperti seperti Prim, Kruskal, dan Warshal algoritma. Jika ada jalan dari simpul x ke simpul y berarti y transitif tergantung pada x. SP CITY S SP 1 - 1 1 CITY - 1 1 - 1 S - 1 1 1 Gambar 2: Determinan ketergantungan transitif kunci DM dari Gambar 2 diperbarui sebagai berikut untuk mencerminkan semua dependensi termasuk yang diperoleh melalui prosedur Ketergantungan-penutupan. S P CITY STATUS JUMLAH SP 2 2 S S 1 CITY 0 0 2 1 0 S 2 0 1 CITY 0 Gambar 3: penutupan Ketergantungan matriks Sekarang saatnya untuk mengganti dependensi langsung, yang mungkin telah menghilang dengan menerapkan dependensi transitif. Namun, algoritma FindOne tidak menemukan apapun memudar ketergantungan. Oleh karena itu, Gambar 3 menunjukkan ketergantungan set optimal. Entri dengan nilai 1 adalah mengidentifikasi komponen dari himpunan ini. Kita sekarang dalam posisi untuk mendapatkan kunci kandidat. A

Halaman 7

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 65 http://www.euroasiapub.org kunci kandidat adalah ofattributes set yang semua atribut lain tergantung. Dari DM akhir kita melihat bahwa SP memiliki properti ini. Diusulkan proses 2NF dan 3NF Normalisasi ini membuat penggunaan kedua ketergantungan dan penentu ketergantungan transitif kunci. Untuk melanjutkan dengan yang 2NF, diasumsikan bahwa tabel sudah dalam bentuk 1NF. Dihasilkan 1NF Relasi adalah: SP\_Relation: {SP, City, Status, Jumlah} Tujuannya adalah untuk menemukan semua dependensi parsial. Untuk menghasilkan bentuk 2NF, kita harus menemukan semua dependensi parsial. Untuk melakukannya, DM dipindai baris demi baris (mengabaikan primary key baris), mulai dari kunci penentu pertama baris yang dipindai sama dengan 2 dan nilai dari kolom yang sesuai dari kunci kandidat sama dengan 2, maka sebagian ketergantungan ditemukan. Dalam Gambar 3, ofCity ketergantungan kepada SP parsial. baris. Oleh karena itu, kita harus membuat tabel baru. Dalam Gambar 4, matriks DM dibagi menjadi dua DMS baru sesuai dengan tabel baru. S P JUMLAH SP 2 2 1 (A) SP\_relation: {SP, Quantity} S CITY STATUS S 2 1 CITY CITY 0 2 1 (B) S\_relation: {S, City, Status} Gambar 4: database dinormalisasi sampai 2NF Dalam rangka untuk mengubah hubungan menjadi 3NF, masing-masing DM adalah scan baris demi baris mulai dari baris pertama. Jika kunci penentu ditemui yang ketergantungan yang tidak parsial (dari Gambar 4) atau itu sepenuhnya tergantung pada bagian dari primary key tabel terpisah harus terbentuk. Tentu saja, jika sebuah tabel sebelumnya membentuk duplikat tidak dihasilkan. Tabel baru ini akan mencakup tombol penentu dan semua atribut lainnya, yang transitif, tergantung pada tombol ini. S P JUMLAH SP 2 2 1 (A) S CITY S 2 1 (B)

Halaman 8

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 66 http://www.euroasiapub.org CITY STATUS CITY 2 1 (C) Gambar 5: database Normalized hingga 3NF Untuk hubungan dengan hanya satu kunci kandidat, 3NF dan BCNF adalah setara. 2.3 Pendekatan normalisasi otomatis menggunakan algoritma paralel Sementara ada algoritma sekuensial biasanya banyak memakan waktu, terutama Proses transformasi relasi menjadi 3NF, sehingga algoritma paralel diusulkan untuk otomatis normalisasi database diberikan. Algoritma yang diusulkan telah diperiksa dengan MPI dan hasil pelaksanaannya pada EDM menunjukkan bahwa pendekatan paralel mengurangi waktu, efisien [11]. Memanfaatkan prosesor p telah mengurangi waktu Database Otomatis Normalisasi untuk c p m n. 2 di mana c adalah overhead komunikasi antara prosesor, m adalah jumlah kunci sederhana, dan n adalah tombol angka ofdeterminant. KESIMPULAN Seperti yang kita lihat dalam pendekatan tradisional siswa perlu mengingat definisi abstrak bentuk normal, apalagi di beberapa tempat penutup kanonik seharusnya tahu. Ini Pendekatan juga memerlukan mengetahui kunci utama dengan menerapkan algoritma, yang sekali lagi adalah pekerjaan yang sulit bagi siswa. Kemudian pendekatan otomatis menggunakan kedua sekuensial dan paralel Algoritma diberikan. Proses ini didasarkan pada generasi ketergantungan matriks, diarahkan matriks grafik, dan penentu utama ketergantungan transitif matriks. Keuntungan dari pendekatan ini lebih dari pendekatan tradisional Primary Key secara otomatis diidentifikasi untuk setiap tabel terakhir yang dihasilkan. Hal ini sangat banyak memakan waktu untuk menggunakan teknik otomatis untuk melakukan hal ini analisis data, sebagai lawan melakukannya secara manual. Pada saat yang sama, proses berurutan adalah diuji dapat diandalkan dan benar . MPI pelaksanaan (algoritma paralel untuk otomatis normalisasi database) hasil pada cluster dengan delapan prosesor menunjukkan cukup pengurangan waktu proses normalisasi database otomatis. Semua ini algoritma sangat efisien. Namun, saya akan membandingkan algoritma ini dengan algoritma lain yang serupa, di masa depan. REFERENSI 1 M Arenas, L Libkin, Pendekatan Informasi-teoritik untuk Bentuk normal untuk Relational dan XML Data, Jurnal ACM (JACM), Vol. 52 (2), hlm. 246-283, 2005

Halaman 9

IJREAS Volume 2, Edisi 2 (February2012) ISSN: 2249-3905 International Journal of Research in Mesin & Terapan 67 http://www.euroasiapub.org 2 Kolahi, S., Ketergantungan-Melestarikan Normalisasi Relational dan XML Data, JournalofComputer Sistem Ilmu, Vol. 73 (4):. Pp 636-647, 2007. 3 Ramez Elmasri dan Shamkant B Navathe, Fundamentals Of Sistem Database, 5 Edition, pg-358 4. Mora, A., M. Enciso, P. Cordero, IP de Guzman, An Preprocessing Efisien Transformasi untuk Dependensi Fungsional Sets berdasarkan Pergantian Paradigma, CAEPIA2003, pp.136-146 2003. 5. Kung, H. dan T. Kasus, Tradisional dan Alternatif database Normalisasi Teknik: Dampaknya terhadap IS / IT Persepsi Mahasiswa dan Kinerja, International Journal of Information Technology Pendidikan, Vol.1, No.1 pp. 53- 76,2004 6 Ramez Elmasri dan Shamkant B Navathe, Fundamentals Of Sistem Database, 5 Edition, pg-392 7 Tanggal, CJ (2000). Sebuah Pengantar Sistem Basis Data 7 "Ed Reading, MA.: Addison-Wesley 8 Yazici, A., dan Z. Karakaya, Normalizing Relational Database Schemas Menggunakan Mathematica, LNCS, Springer-Verlag, Vol.3992, pp. 375-382, 2006. 9. Akehurst, DH, B. Bordbar, PJ Rodgers, dan NTGDalgliesh, otomatis Normalisasi via Metamodelling, ASE 2002 Workshop deklaratif Meta Pemrograman untuk Mendukung Pengembangan Perangkat Lunak, 2002 10 AH Bahmani, M. Naghibzadeh, dan B. Bahmani. Normalisasi basis data otomatis dan pembangkitan kunci primer. IEEE CCECE / CCGEI, halaman 11-16, Mei 2008 11. Amir -H. Bahmani, S.Kazem Shekofteh, Mahmoud Naghibzadeh, Hossein Deldari. Algoritma Paralel untuk Automatic database Normalisasi. IEEE / ICCAE, Publikasi, Tahun: 2010, Page (s): 157-161