

PENGUJIAN MODEL *COST ANOMALI* BASIS DATA RELASIONAL BENTUK NORMAL KEDUA MENGGUNAKAN SAS PROGRAMMING

Siti Mardiana

Program Studi Sistem Informasi, Universitas Widyatama
Jalan Cikutra no. 204 A Bandung Telp. : (022) 727 5855
E-mail : siti.mardiana@widyatama.ac.id

Abstrak

Denormalisasi sering dilakukan terhadap basis data relasional dengan alasan untuk peningkatan performansi. Namun keuntungan yang didapat seringkali harus dibayar dengan harga tertentu, yaitu munculnya anomali. Paper ini membahas tentang penghitungan ekspektasi cost untuk basis data yang telah di-denormalisasi dari bentuk normal ketiga menjadi bentuk normal kedua. Cost dihitung berdasarkan effort yang harus dikeluarkan oleh sistem basis data untuk mencegah terjadinya anomali jika dilakukan operasi update, insert, dan delete. Pengujian model cost anomali dilakukan terhadap skema basis data relasional dengan kardinalitas N:1. Pengujian menggunakan aplikasi pemrograman SAS untuk melihat kecenderungan kurva probabilitas cost anomali. Dari pengujian yang telah dilakukan bisa disimpulkan bahwa kurva cost anomali menunjukkan angka yang semakin besar dengan meningkatnya volume denormalisasi suatu tabel.

Kata kunci : basis data relasional, denormalisasi, anomali, cost anomaly, SAS programming

1. PENDAHULUAN

Model relasional merupakan bentuk pemodelan basis data yang paling banyak digunakan dibandingkan dengan jenis pemodelan lain (Hoffer *et al.*, 2007). (Elmasri, Navathe, 2011) dan (Date, 2004) menyatakan bahwa model relational adalah konsep basis data yang sangat penting, bahkan kelahiran model relational disebut sebagai peristiwa terpenting dalam sejarah basis data.

Salah satu alasan mengapa model relasional banyak digunakan adalah karena model relasional memenuhi persyaratan penting untuk sebuah basis data, yaitu menjamin adanya integritas basis data (Hawryskiewycs, 1990). Hal tersebut bisa dicapai oleh basis data relasional melalui fitur yang disebut bentuk normal (*normal form*), yang dinyatakan dalam tingkatan bentuk normal pertama, kedua, ketiga, keempat, dan kelima. Karena bentuk normal keempat dan kelima jarang dipakai dalam aplikasi basis data pada umumnya, maka untuk seterusnya dalam paper ini pembahasan bentuk normal tertingi mengacu pada bentuk normal ketiga.

Bentuk normal menyatakan tingkat redundansi dari sebuah tabel basis data. Semakin tinggi bentuk normal sebuah tabel maka akan semakin rendah tingkat redundansinya. Tingkat redundansi yang rendah menunjukkan semakin tingginya jaminan integritas basis data. Artinya, jika sebuah data hanya tersimpan satu kali pada satu lokasi (tidak ada redundansi), maka jika dilakukan operasi *update* atau *delete* terhadap data itu, maka basis data tetap valid, akurat dan konsisten.

Bentuk normal yang lebih tinggi biasanya dicapai dengan melakukan pemecahan relasi atau dekomposisi terhadap sebuah tabel relasi. Proses dekomposisi mengubah sebuah tabel bentuk normal rendah menjadi beberapa tabel dalam bentuk normal yang lebih tinggi. Setiap tabel baru hasil dekomposisi secara umum mempunyai ukuran fisik yang lebih kecil dari pada tabel semula.

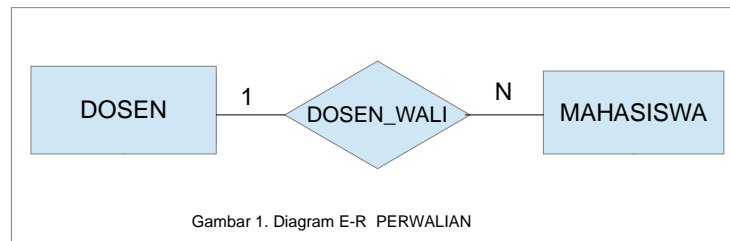
Pada tahap implementasi, setiap tabel relasi akan menjadi sebuah file data. Pada saat operasi *retrieve* terhadap basis data, beberapa file akan digabung dengan operasi *join* untuk menghasilkan informasi yang diinginkan. Operasi *join* sangat memberatkan terhadap basis data karena harus membuka beberapa file secara bersamaan serta melakukan pencarian di setiap file untuk menemukan *record* dengan kriteria yang dimaksud. Hal ini sangat berpengaruh terhadap performansi basis data. Oleh sebab itu, dalam implementasinya, tabel bentuk normal ketiga seringkali dikembalikan menjadi bentuk normal yang lebih rendah, misal bentuk normal kedua. Proses ini disebut dengan denormalisasi.

Dengan pemakaian bentuk normal kedua akan diperoleh benefit berupa performansi yang meningkat. Namun performansi yang meningkat tersebut harus dibayar dengan munculnya redundansi (Coronel, *et al.* 2013). Redundansi ini yang akan mengakibatkan timbulnya anomali ketika dilakukan operasi *update* dan *insert*. Untuk menghindari anomali tersebut maka sistem basis data harus mengeluarkan usaha tertentu untuk menjaga agar tabel tetap akurat dan konsisten meskipun tabel tersebut dalam bentuk normal kedua. Dalam paper ini akan dilakukan pengujian untuk melihat seberapa besar usaha yang harus dikeluarkan (dinyatakan dalam ekspektasi

cost) untuk mencegah terjadinya anomali. Model yang diuji merupakan model penghitungan cost anomali yang dikemukakan oleh Westland (Westland, 1992).

2. TINJAUAN PUSTAKA

Misal sebuah basis data PERWALIAN dengan diagram Entiti-Relasi sebagai berikut:



Entiti DOSEN mempunyai atribut : KODE_DSN, NAMA_DSN, ALAMAT_DSN, dimana KODE_DSN merupakan PRIMARY KEY. Entiti MAHASISWA mempunyai atribut : KODE_MHS, NAMA_MHS, ALAMAT_MHS, dimana KODE_MHS merupakan PRIMARY KEY. Dalam perancangan fisik (*physical design*), diagram E-R yang terlihat pada Gambar 1 akan diimplementasikan menjadi tiga tabel, yaitu:

Tabel DOSEN

| | | |
|-----------------|----------|------------|
| <u>KODE_DSN</u> | NAMA_DSN | ALAMAT_DSN |
|-----------------|----------|------------|

Tabel MAHASISWA

| | | |
|-----------------|----------|------------|
| <u>KODE_MHS</u> | NAMA_MHS | ALAMAT_MHS |
|-----------------|----------|------------|

Tabel DOSEN_WALI

| | |
|-----------------|-----------------|
| <u>KODE_DSN</u> | <u>KODE_MHS</u> |
|-----------------|-----------------|

Masing-masing tabel di atas berada dalam bentuk normal ketiga karena semua atribut yang bukan kunci (*non primary key*) hanya bergantung kepada atribut kunci.

Jika dari skema basis data di atas diinginkan daftar nama mahasiswa yang menjadi mahasiswa wali dari setiap dosen (dengan menampilkan nama dosen), maka harus dilakukan operasi *join* terhadap tiga tabel, yaitu DOSEN, MAHASISWA, dan DOSEN_WALI, untuk menghasilkan query yang dimaksud. Dalam *structured query language* (SQL) akan dinyatakan dengan *syntax* sebagai berikut:

```

Select NAMA_MHS, NAMA_DSN
From DOSEN, MAHASISWA, DOSEN_WALI
Where DOSEN_WALI.KODE_MHS=MAHASISWA.KODE_MHS and
      DOSEN_WALI.KODE_DSN=DOSEN.KODE_DSN
    
```

Dasar operasi *join* seperti di atas adalah operasi *cartesian product* yang diikuti dengan operasi *select* (Misra, Eich, 1992). *Cartesian product* adalah operasi basis data yang mahal (Zhou, 2007). (Schmidt, *et al.*, 2008) menyatakan bahwa operasi *join* adalah operasi yang sangat mahal untuk basis data karena tabel hasil *join* akan berukuran sangat besar sehingga mempengaruhi performansi basis data.

1.1. Denormalisasi

Query di atas akan lebih mudah dilakukan jika tabel basis data tidak dalam bentuk normal ketiga sehingga tidak lagi diperlukan operasi *join*. Misal ketiga tabel DOSEN, MAHASISWA, dan DOSEN_WALI digabung (melalui proses denormalisasi) sehingga hanya menjadi satu tabel, misal diberi nama tabel PERWALIAN:

Tabel PERWALIAN

| | | | | | |
|-----------------|----------|------------|----------|----------|------------|
| <u>KODE_MHS</u> | NAMA_MHS | ALAMAT_MHS | KODE_DSN | NAMA_DSN | ALAMAT_DSN |
|-----------------|----------|------------|----------|----------|------------|

Tabel di atas tidak berada dalam bentuk normal ketiga karena terdapat atribut yang tergantung pada atribut selain primary key. Dalam tabel di atas, KODE_MHS adalah *primary key*, sedangkan NAMA_DSN bergantung pada KODE_DSN, dan bukan pada KODE_MHS. Query SQL sebelumnya bisa dilakukan dengan lebih singkat hanya dengan operasi *select*:

```

Select NAMA_MHS, NAMA_DSN
From PERWALIAN
    
```

1.2. Konsep *Left-hand side (LT)* dan *Right-hand side (RT)* dari sebuah tabel

Dalam model penghitungan *cost* anomali oleh Westland (Westland, 1992) dipakai konsep *left-hand side* dan *right-hand side* untuk menggambarkan ketergantungan antar informasi di dalam tabel hasil denormalisasi. Pada tabel PERWALIAN terlihat bahwa atribut NAMA_MHS dan ALAMAT_MHS tergantung pada KODE_MHS, sedangkan NAMA_DSN dan ALAMAT_DSN tergantung pada KODE_DSN. Dalam notasi ketergantungan fungsional dinyatakan sebagai berikut:

$$\text{KODE_MHS} \rightarrow \text{NAMA_MHS, ALAMAT_MHS} \quad (1)$$

$$\text{KODE_DSN} \rightarrow \text{NAMA_DSN, ALAMAT_MHS} \quad (2)$$

Notasi ketergantungan fungsional digambarkan pada skema tabel PERWALIAN sebagai berikut:

Tabel PERWALIAN

| | | | | | |
|----------|----------|------------|----------|----------|------------|
| KODE_MHS | NAMA_MHS | ALAMAT_MHS | KODE_DSN | NAMA_DSN | ALAMAT_DSN |
| (1) | | | (2) | | |

Dalam kasus tabel PERWALIAN dinyatakan bahwa seorang dosen yang tidak mempunyai mahasiswa wali tidak akan masuk ke dalam tabel tersebut. Dengan kata lain, keberadaan dosen wali di tabel tersebut ditentukan oleh keberadaan mahasiswa wali. (Westland,1992) menyatakan bahwa pada kasus seperti di atas, maka bagian (1) adalah *Left-hand side (LT)* dan bagian (2) adalah *Right-hand side (RT)* dari tabel PERWALIAN, dengan notasi ketergantungan fungsional ditulis sebagai berikut:

$$\text{LT} \rightarrow \text{RT}$$

1.3. Model Ekspektasi *Cost Anomali* (Westland, 1992)

Model ekspektasi *cost* nomali yang ditawarkan Westland didasarkan pada pemahaman bahwa anomali hanya akan muncul jika terjadi peristiwa di dunia nyata yang menyebabkan perubahan pada basis data. Dalam kasus tabel PERWALIAN, peristiwa tersebut misalnya adalah adanya mahasiswa baru yang harus dimasukkan ke dalam basis data. Untuk menggambarkan proses yang menyebabkan perubahan dalam basis data, Westland menggunakan model *renewal process* yang dinyatakan dengan proses Poisson. Model *renewal process* digunakan untuk mensimulasikan proses penyisipan *tuple* baru ke dalam basis data.

2.3.1 Model *renewal process* (Karlin, Taylor, 1975)

Proses Poisson $\{N(t), t \geq 0\}$ merupakan *renewal process* yang mempunyai distribusi antar kejadian:

$$F(x) = 1 - e^{-\lambda x}, \quad x \geq 0$$

Peluang $N(t)$ terjadi dalam perioda t dinyatakan dengan :

$$\Pr\{N(t)=k\} = \frac{(\lambda t)^k e^{-\lambda t}}{k!} \quad (1)$$

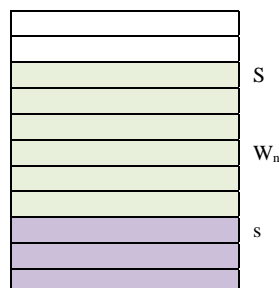
$$\text{dengan ekspektasi } E[N(t)] = M(t) = \lambda t \quad (2)$$

Parameter λ dapat dihitung dari jumlah transaksi pada suatu perioda.

2.3.2 Model Inventori

Secara teoritis, kemungkinan jumlah mahasiswa baru yang akan masuk ke dalam basis adalah tak terbatas. Namun dalam pemodelan ini, "tak terbatas" tadi dibatasi oleh kemampuan daya simpan basis data. Untuk menggambarkan proses tersebut, Westland menggunakan model inventori yang dijelaskan di bawah.

Misal S adalah tingkat persediaan dalam posisi aman (*safety stock*), dan s menunjukkan posisi persediaan dalam keadaan kritis, maka berlaku $S > s$ dimana S dan s bilangan positif.



Gambar 2 : Model Inventori

Dalam model inventori, adanya permintaan (*demand*) mengakibatkan status inventori turun dari posisi S menuju ke s . Jika status persediaan dalam posisi s (posisi kritis) maka persediaan akan ditambah hingga mencapai posisi S . Jika kondisi ini dihubungkan dengan *renewal process* $N(t)$, maka $N(S-s)+1$ adalah jumlah perioda

permintaan yang terlewat (dimana status inventori berkurang dari S menuju s) sampai order yang pertama untuk mengisi kembali inventori.

Jika W_n adalah tingkat persediaan pada akhir periode permintaan ke n , maka *conditional distribution function* adalah fungsi distribusi terbatas tingkat persediaan dalam perioda dimana permintaan (*demand*) akan langsung dipenuhi (karena posisi inventori di atas s , artinya tidak kritis). Berdasarkan persamaan *renewal process* maka diperoleh:

$$\lim_{n \rightarrow \infty} \Pr(S - x \leq W_n) = \frac{M(x)}{1 + M(S - s)} \quad (3)$$

dimana x adalah jumlah *demand* dalam periode tertentu. Analoginya di dalam basis data, jika ruang penyimpanan di dalam basis data tidak penuh, maka penyisipan akan langsung dilakukan. Namun jika ruang penyimpanan di dalam basis data penuh, maka penyisipan tidak bisa dilakukan. Jika mengikuti analogi model inventori, maka sebagian *record* harus dihapus.

Pada periode ke n , W_n identik dengan volume basis data aktual θ , sehingga persamaan (3) bisa dinyatakan sebagai berikut:

$$\Pr\{\theta \leq s + x | \theta \leq S\} = \frac{M(x)}{1 + M(S - s)}$$

Substitusi dengan persamaan (2) diperoleh:

$$\Pr\{\theta \leq s + x | \theta \leq S\} = \frac{\lambda t(x)}{1 + \lambda t(S - s)}, \quad x \in [0, S - s]$$

Sehingga ekspektasi harga basis data aktual θ adalah:

$$E(\theta | \theta \leq S) = s + \frac{\lambda t(S - s)^2}{2 + 2\lambda t(S - s)} \quad (4)$$

2.3.3 Model Ekspektasi Cost Anomali Perubahan

Misal tabel PERWALIAN berisi data sebagai berikut:

Tabel 1. Tabel PERWALIAN

| KODE_MHS | NAMA_MHS | ALAMAT_MHS | KODE_DSN | NAMA_DSN | ALAMAT_DSN |
|----------|----------|------------------|----------|----------|---------------|
| M00001 | Abe | Jl. Nias 17 | D001 | Ratna | Jl. Jalak 20 |
| M00002 | Adi | Jl. Beo 50 | D001 | Ratna | Jl. Jalak 20 |
| M00003 | Bela | Jl. Sarijadi 21 | D001 | Ratna | Jl. Jalak 20 |
| M00005 | Betty | Jl. Kemangi 34 | D002 | Dewi | Jl. Merapi 65 |
| M00006 | Desi | Jl. Mawar 2 | D002 | Dewi | Jl. Merapi 65 |
| M00007 | Fadli | Jl. Ciliwung 27 | D003 | Rosa | Jl. Pahlwan 6 |
| M00008 | Gina | Jl. Salak 5 | D004 | Ema | Jl. Garuda 44 |
| M00009 | Yana | Jl. Prambanan 18 | D004 | Ema | Jl. Garuda 44 |

LT
RT

Misal dosen Ratna pindah alamat dari Jl. Jalak 20 ke Jl. Samudra 34, harus dilakukan perubahan pada basis data tersebut. Untuk menghindari terjadinya anomaly, maka perubahan data dosen [D001, Ratna, Jl. Jalak 20] menjadi [D001, Ratna, Jl. Samudra 34] akan mengikuti langkah sebagai berikut:

1. memeriksa setiap *tuple* pada LT untuk memeriksa apakah RT-nya mengandung atribut $KODE_DSN = "D001"$
2. jika benar, maka *update* RT, dengan mengubah atribut alamat dosen menjadi "Jl. Samudra 32"

Jika jumlah RT untuk suatu LT sangat besar, maka besar pula usaha yang harus dilakukan untuk mencegah anomaly. Rekapitulasi ranking jumlah LT per RT dari tabel PERWALIAN adalah sebagai berikut:

| KODE_DSN | NAMA_DSN | ALAMAT_DSN | Jumlah LT | Ranking |
|----------|----------|---------------|-----------|---------|
| D001 | Ratna | Jl. Jalak 20 | 3 | 1 |
| D004 | Ema | Jl. Garuda 44 | 2 | 2 |
| D002 | Dewi | Jl. Merapi 65 | 2 | 2 |
| D003 | Rosa | Jl. Pahlwan 6 | 1 | 3 |

RT

Tabel 2. Ranking RT untuk tabel PERWALIAN

Distribusi acak jumlah LT per RT dinyatakan dengan model distribusi *rank-frequency* Zipfian:

$$\Pr\{t = i | \tau\} = \frac{i^{-\tau}}{\zeta(\tau)} \quad i=1,2,\dots \quad \tau \in (1,2) \quad (5)$$

Parameter t adalah variable acak yang menunjukkan ranking RT. Ranking RT ditentukan oleh jumlah frekuensi LT untuk tiap RT. Ranking $RT=1$ ($t=1$) menunjukkan RT dengan frekuensi LT yang tertinggi. $\zeta(\tau)$ adalah fungsi Riemann-Zeta yang digunakan untuk menentukan seberapa besar kemungkinan suatu tuple bergabung dalam basis data yang dinyatakan dalam persamaan:

$$\zeta(\tau) = \sum_{n=1}^{\infty} \frac{1}{n^{\tau}} \quad (6)$$

Harga τ antara 1 dan 2 ditentukan berdasarkan analogi pada fenomena *temperature of discourse* (Mandelbrot, 1983). Harga $\tau \leq 1$ menunjukkan ketergantungan antar informasi yang lemah, sedangkan harga $\tau \geq 2$ menunjukkan abnormalitas informasi. Sedangkan harga τ antara 1 dan 2 menunjukkan ketergantungan yang kuat antar informasi. Fenomena ini digunakan untuk menggambarkan probabilitas suatu informasi untuk masuk ke dalam basis data. Hanya informasi dengan ketergantungan kuat yang akan masuk ke dalam basis data.

Jika jumlah aktual basis data adalah θ , maka urutan jumlah LT per RT dari yang terbesar ke yang terkecil adalah:

$$\frac{\theta}{\zeta(\tau)} t^{-\tau} \quad t=1, \dots, t_{\min}(\theta) \quad (7)$$

$t_{\min}(\theta)$ adalah ranking RT terendah (≥ 1) karena memiliki jumlah LT paling kecil, dinyatakan dengan persamaan:

$$t_{\min}(\theta) = \inf \left\{ t \mid \frac{\theta}{\zeta(\tau)} t^{-\tau} \geq 1 \right\} \quad (8)$$

Ekspektasi harga LT per RT dinyatakan dengan persamaan :

$$\frac{\theta}{\zeta(\tau)} \sum_{t=1}^{t_{\min}} t^{-2\tau} \quad (9)$$

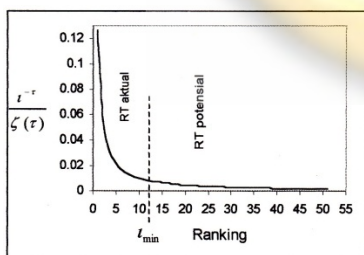
Jika ψ adalah *cost* untuk menghilangkan satu anomali karena operasi *update* dan v adalah jumlah transaksi *update* dalam periode t , maka ekspektasi *cost* untuk mencegah terjadinya anomali untuk operasi *update* dalam periode t adalah:

$$\psi v \frac{\theta}{\zeta(\tau)} \sum_{t=1}^{t_{\min}} t^{-2\tau} \quad (10)$$

2.3.4 Model Ekspektasi Cost Anomali Penyisipan

Anomali penyisipan terjadi jika sebuah data dari suatu entiti akan disisipkan ke dalam tabel, namun pasangan datanya tidak tersedia (Philip, 2007). Dalam model ekspektasi *cost* anomali yang ditawarkan oleh Westland (Westland, 1992), pernyataan tersebut diimplementasikan dengan pasangan data LT-RT, artinya operasi *insert* sebuah LT ke dalam tabel hasil denormalisasi akan menyebabkan anomali jika LT tersebut tidak mempunyai RT yang berhubungan. Seperti dijelaskan sebelumnya bahwa secara teoritis LT dan RT berjumlah tak terhingga.

Dari persamaan (8) terlihat t_{\min} adalah RT terakhir yang masuk ke dalam tabel basis data, sehingga RT potensial yang akan masuk ke dalam basis data adalah RT yang berada pada $t > t_{\min}$.



Gambar 3. RT potensial berada pada ekor kurva $> t_{\min}$

Kembali ke persamaan (5) yang merupakan distribusi acak LT per RT (*rank-frequency* Zipfian) yang jika digabung dengan *renewal process* persamaan (2) akan diperoleh proses Poisson dengan parameter μ sebagai berikut:

$$\mu_t = \lambda \frac{t^{-\tau}}{\zeta(\tau)} \quad t=1,2,\dots \quad \tau \in (1,2) \quad (11)$$

Karena proses penyisipan berkaitan dengan RT potensial yang dimulai dari $t=t_{\min}$, maka dari persamaan (8) didapat

$$t_{\min} \approx \left[\frac{\zeta(\tau)}{\theta} \right]^{\frac{1}{\tau}}$$

Jika persamaan di atas disubstitusikan ke dalam persamaan(11) maka akan menjadi :

$$\mu_0 = \lambda \frac{t_{min}^{-\tau}}{\zeta(\tau)}$$

Westland menyatakan bahwa jumlah terjadinya anomali penyisipan dalam periode tertentu dinyatakan dengan persamaan:

$$\int_0^{\infty} n \left[\frac{1 - \Gamma_1(2, \mu_0 t)}{t} \right]^n dn$$

yang dijabarkan menjadi: $\frac{1 - \Gamma_1(2, \mu_0 t)}{t} \left[\ln^{-2} \left[\frac{1 - \Gamma_1(2, \mu_0 t)}{t} \right] - \ln^{-1} \left[\frac{1 - \Gamma_1(2, \mu_0 t)}{t} \right] \right]$

$\Gamma_1(2, \mu_0 t)$ adalah fungsi gama tidak lengkap $\Gamma(a, x)$ dimana $a=2$ dan $x=\mu_0 t$.

Fungsi gama tidak lengkap $\Gamma(a, x)$ dinyatakan dengan : $\Gamma(a) - x^a e^{-x} \sum_{n=0}^{\infty} \frac{x^n}{\Gamma(a+n+1)}$

Karena harga cost anomali penyisipan hanya tergantung pada jumlah terjadinya anomali penyisipan maka persamaan ekspektasi cost anomali penyisipan adalah:

$$\alpha \int_0^{\infty} n \left[\frac{1 - \Gamma_1(2, \mu_0 t)}{t} \right]^n dn \tag{12}$$

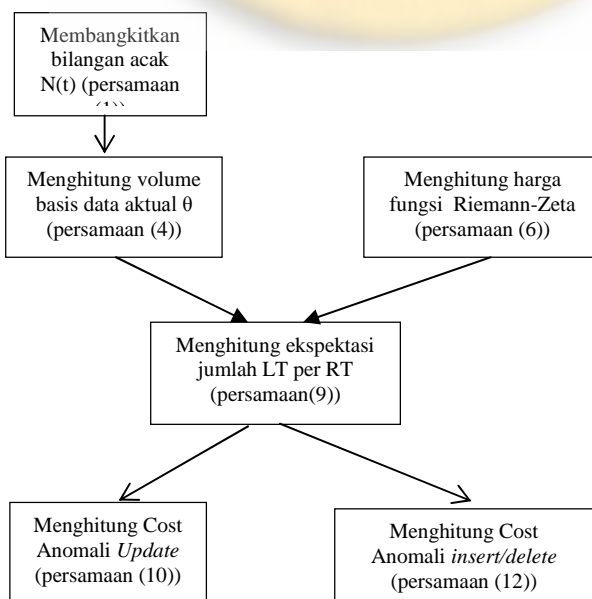
Dalam persamaan (12) di atas terdapat parameter α yang tidak dibahas dalam model Westland. Parameter α adalah cost untuk satu anomali penyisipan.

2.3.5 Model Penghitungan cost anomali penghapusan

Westland menyatakan bahwa pada prinsipnya proses terjadinya anomali penghapusan dan penyisipan adalah sama. Jika anomali penyisipan terjadi ketika terdapat LT yang akan masuk ke dalam basis data namun RT-nya tidak tersedia, maka anomali penghapusan terjadi sebaliknya: terdapat LT yang akan keluar dari basis data, namun RT yang bersangkutan harus berada dalam basis data, atau informasi tentang RT tersebut hilang. Dengan demikian maka harga cost anomali penghapusan akan sama dengan cost anomali penyisipan.

3. METODE PENELITIAN

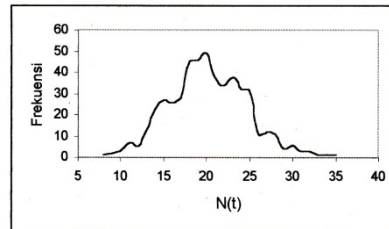
Pengujian model cost anomali Westland menggunakan pemrograman SAS dengan mengikuti alur diagram berikut:



4. HASIL DAN PEMBAHASAN

4.1. Membangkitkan bilangan acak $N(t)$

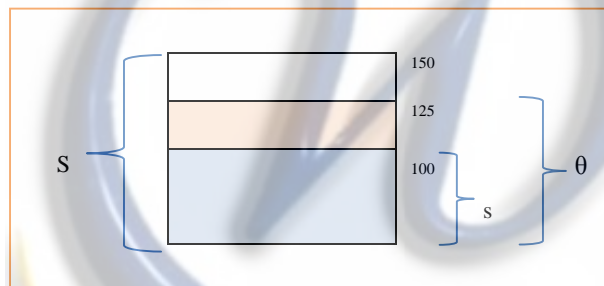
Bilangan acak $N(t)$ digunakan untuk mensimulasikan rata-rata jumlah transaksi basis data (berupa *update* atau *insert*) pada suatu periode tertentu. Dalam pengujian ini, data acak yang dibangkitkan sebanyak 1000 dan berdistribusi Poisson dengan $\lambda_p=20$ dari populasi $P=1000000$. Diagram data acak yang dihasilkan terlihat pada Gambar 4.



Gambar 4. Sebaran data acak

4.2. Menghitung Nilai Basis Data Aktual θ

Untuk pengujian ini, ditentukan dipilih kapasitas penyimpan $S=150$, dan kapasitas minimum basis data $s=100$. Maka dari perhitungan persamaan (4) diperoleh harga $\theta=124,9855$ (atau dibulatkan menjadi 125). Nilai basis data aktual θ digambarkan sebagai berikut:



Gambar 5. Kapasitas basis data sebagai implementasi dari model inventori

4.3. Menghitung Fungsi Riemann-Zeta

Seperti telah dibahas dalam landasan teori bahwa fungsi Riemann-Zeta digunakan untuk mensimulasikan kecenderungan suatu informasi untuk bergabung dalam basis data. Hasil penghitungan fungsi Riemann-Zeta adalah sebagai berikut:

Tabel 3. Output fungsi Riemann-Zeta

| T | $\zeta(\tau)$ |
|-----|---------------|
| 1 | 15,0859 |
| 1,1 | 7,89228 |
| 1,2 | 5,22919 |
| 1,3 | 3,86691 |
| 1,4 | 3,09241 |
| 1,5 | 2,60955 |
| 1,6 | 2,28513 |
| 1,7 | 2,05414 |
| 1,8 | 1,88201 |
| 1,9 | 1,74974 |
| 2 | 1,64491 |

Hasil fungsi Riemann-Zeta akan dijadikan sebagai penyebut dalam persamaan (9) untuk menghitung distribusi LT per RT, sehingga harga $\zeta(\tau)$ yang besar (τ kecil) akan menghasilkan jumlah LT per RT yang kecil. Semakin kecil harga τ maka semakin kecil jumlah LT per RT.

4.4. Menentukan ekspektasi jumlah LT per RT

Untuk setiap nilai $\zeta(\tau)$ pada Tabel 3 kemudian dilakukan penghitungan ekspektasi jumlah LT per RT dengan hasil sebagai berikut:

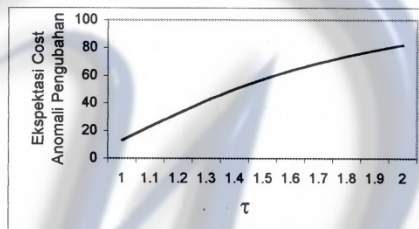
Tabel 4. Ekspektasi jumlah LT per RT

| T | $\zeta(\tau)$ | Ekspektasi LT per RT |
|-----|---------------|----------------------|
| 1 | 15,0859 | 12,9658 |
| 1,1 | 7,89228 | 22,9684 |
| 1,2 | 5,22919 | 32,5672 |
| 1,3 | 3,86691 | 41,8409 |
| 1,4 | 3,09241 | 50,1635 |
| 1,5 | 2,60955 | 57,4202 |
| 1,6 | 2,28513 | 63,7213 |
| 1,7 | 2,05414 | 69,2241 |
| 1,8 | 1,88201 | 74,0780 |
| 1,9 | 1,74974 | 78,3749 |
| 2 | 1,64491 | 82,2258 |

Dari tabel di atas terlihat bahwa ekspektasi LT per RT membesar seiring dengan besarnya harga τ .

4.5. Menghitung cost anomali *update*

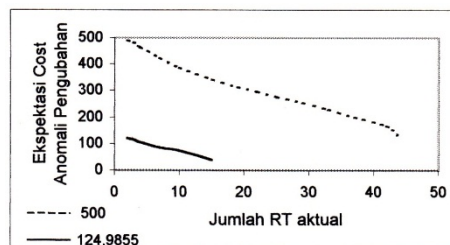
Penghitungan *cost anomaly update* (persamaan (10)) didasarkan pada harga ekspektasi LT per RT. Dalam model ini, harga parameter ψ tidak dibahas, sehingga dalam pengujian yang telah dilakukan harga ψ ditetapkan sama dengan 1. Dengan demikian harga cost anomali hanya dipengaruhi oleh ekspektasi jumlah LT per RT. Hasil penghitungan akan tampak sama seperti Tabel 4 dengan tampilan diagram sebagai berikut:



Gambar 6. Ekspektasi cost anomaly pengubahan berdasarkan harga τ

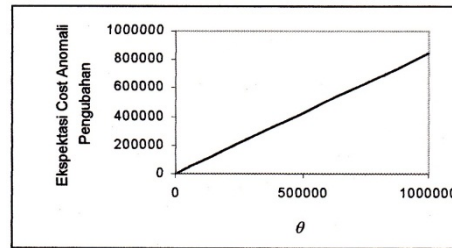
Dari Gambar 6 terlihat bahwa cost anomali akan semakin tinggi dengan semakin tingginya harga τ . Harga τ yang tinggi menunjukkan semakin tingginya jumlah LT per RT. Jika diterapkan kepada tabel PERWALIAN maka jika rata-rata jumlah mahasiswa untuk setiap dosen adalah besar maka cost anomali untuk tabel tersebut juga besar. Artinya, cost anomali paling rendah akan dicapai jika kardinalitas antara LT dan RT adalah 1:1 (setiap dosen hanya mempunyai satu mahasiswa wali).

Pengujian cost anomali pengubahan juga dilakukan dengan parameter RT aktual, sekaligus dengan mengubah harga kapasitas basis data θ dari 124, 9855 ke 500. Hasil pengujian terlihat pada Gambar 7. Pada gambar tersebut terlihat bahwa jika jumlah RT aktual bertambah maka *cost* anomali pengubahan akan menurun. Hal tersebut karena jika jumlah LT tetap dan jumlah RT semakin banyak (hingga mendekati atau sama dengan jumlah LT), maka kardinalitas LT:RT menjadi 1:1. Sebuah tabel hasil denormalisasi yang mempunyai kardinalitas 1:1 akan mempunyai nilai *cost* anomali yang terendah.



Gambar 7. Ekspektasi *cost* anomali pengubahan dengan harga θ yang berbeda

Pengujian *cost* anomali pengubahan juga dilakukan dengan harga θ yang diperbesar. Hasil pengujian tampak pada Gambar 8. Pada gambar tersebut terlihat bahwa semakin besar ukuran basis data maka semakin besar pula *cost* anomali yang ditimbulkan.



Gambar 8. Ekspektasi cost anomaly dengan θ besar

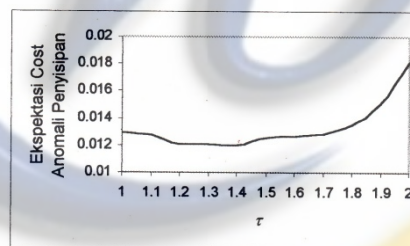
4.6. Menghitung cost anomaly *insert* (penyisipan) dan *delete* (penghapusan)

Dengan memakai parameter $\alpha=1$, $\tau=1,2$, dan $\theta=125$ diperoleh harga ekspektasi jumlah anomaly penyisipan sebagai berikut:

Tabel 5. Ekspektasi jumlah anomaly penyisipan (angka yang ditampilkan adalah pembulatan)

| τ | $\frac{c_{min}}{\zeta(\tau)}$ | Ekspektasi jumlah anomaly penyisipan |
|--------|-------------------------------|--------------------------------------|
| 1 | 0,0083 | 0,0129 |
| 1,1 | 0,0085 | 0,0128 |
| 1,2 | 0,0081 | 0,0121 |
| 1,3 | 0,0081 | 0,0120 |
| 1,4 | 0,0080 | 0,0120 |
| 1,5 | 0,0082 | 0,0125 |
| 1,6 | 0,0082 | 0,0126 |
| 1,7 | 0,0083 | 0,0128 |
| 1,8 | 0,0084 | 0,0135 |
| 1,9 | 0,0088 | 0,0150 |
| 2 | 0,0095 | 0,0180 |

Dari Tabel 5 dihasilkan diagram ekspektasi cost anomaly penyisipan seperti di bawah:



Gambar 9. Diagram ekspektasi cost anomaly penyisipan

Senada dengan anomaly pengubahan, *cost* untuk anomaly penyisipan juga meningkat sejalan dengan meningkatnya harga τ . Artinya, jika suatu RT potensial dibutuhkan oleh banyak LT maka ketidakhadiran RT potensial tersebut akan mengakibatkan *cost* yang semakin besar.

Westland menyatakan bahwa proses penyisipan pada prinsipnya sama dengan proses penghapusan, sehingga *cost* anomal penghapusan akan sama dengan *cost* anomaly penyisipan. Asumsi tersebut mendekati kebenaran jika antara LT dan RT mempunyai kardinalitas 1:1 (artinya jika LT dihapus maka RT juga harus dihapus). Untuk LT dan RT yang mempunyai kardinalitas 1:N maka penghapusan suatu LT tidak akan mengakibatkan hilangnya sebuah RT, karena pada *tuple* yang lain terdapat RT yang sama (sehingga basis data tidak akan kehilangan informasi tentang RT yang bersangkutan).

5. KESIMPULAN

Dari hasil pengujian dan pembahasan yang telah dilakukan maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Untuk mendapatkan performansi dalam basis data relational, maka seringkali dilakukan denormalisasi. Namun keuntungan itu harus dibayar dengan harga tertentu sebagai akibat dari terjadinya anomaly.
2. Model ekspektasi *cost* anomaly yang disarankan oleh Westland bisa menunjukkan secara umum bahwa *cost* akan naik dengan semakin besarnya jumlah LT per RT. *Cost* paling minimum berada pada saat jumlah LT=RT. Artinya, denormalisasi akan menghasilkan *cost* yang paling rendah jika LT dan RT berasal dari tabel yang merupakan implementasi dari dua entiti dengan kardinalitas 1:1. Namun untuk

- penghitungan *cost* secara nominal masih diperlukan penelitian lebih lanjut, terutama menyangkut parameter ψ pada persamaan (10) dan parameter α pada persamaan (12).
3. Pemodelan yang disarankan Westland dicetuskan pada suatu era dimana media penyimpan masih mahal, sehingga Westland memakai model inventori sebagai sarana untuk menggambarkan keluar-masuknya data (dianalogikan dengan keluar-masuknya barang dalam sistem inventori dimana barang harus ada yang terjual agar terdapat tempat kosong untuk barang yang akan masuk). Dalam era sekarang dimana media penyimpan sangat murah, model inventori mungkin tidak relevan lagi karena orang cenderung untuk menambah media penyimpan dari pada menghapus data yang tersimpan.
 4. Pengujian dilakukan dengan berbagai asumsi, misal kapasitas media penyimpan $S=150$, kapasitas minimum media penyimpan $s=100$, dan nilai λ untuk proses Poisson dipakai $\lambda_p=20$. Perlu dilakukan uji coba dengan nilai parameter yang diubah-ubah untuk melihat efektifitas model penghitungan *cost* anomali yang ditawarkan oleh Westland.

DAFTAR PUSTAKA

- Coronel, C., Morris, S., Rob, P. 2013. *Database Systems: Design, Implementation and Management, 10th Edition*. Course Technology Cengage Learning, Massachusetts.
- Elmasri, R., Navathe, S.B. 2011. *Fundamentals of Database Systems, 6th Edition*. Pearson Education Inc., New York.
- Hawryskiewycs, I.T. 1990. *Relational Database Design – An Introduction*, Prentice Hall, Sydney.
- Hoffer, J.A., Prescott, M.B., McFadden, F.R. 2007. *Modern Database Management, 8th Edition*, Pearson Education Inc., New York.
- Karlin, S., Taylor, H.M. 1975. *A first course in stochastic process, 2nd Edition*, Academic Press, New York.
- Mandelbrot, B. 1983. *The Fractal Geometry of Nature*, WH Freeman, New York.
- Misra, P., Eich, M.H. 1992. *Join Processing in Relational Database*, ACM Computing Surveys, 24(1), pages 63-113.
- Philip, G.C. 2007. *Teaching Database Modeling and Design: Areas of Confusion and Helpful Hints*, Journal of Information Technology Education, vol. 6, pages 481-497.
- Westland, J.C. 1992. *Economic incentives for databases normalization*. Information Processing and Management. Vol. 28. Pages 647-662.
- Zhou, Zehai. 2007. *Using Heuristics and Genetic Algorithms for Large-scale Database Query Optimization*. Journal of Information and Computing Science Vol. 2, No. 4, 2007, pp. 261-280