



Analisis kinerja pemodelan data *star schema* pada data perpustakaan

Dini Nurmalasari¹, Mutia Sari Zulvi² dan Puja Hanifah³

¹Politeknik Caltex Riau, email: dini@pcr.ac.id

²Politeknik Caltex Riau, email: mutia@pcr.ac.id

³Politeknik Caltex Riau, email: puja@pcr.ac.id

Abstrak

Sistem informasi perpustakaan Politeknik Caltex Riau yang datanya mulai terorganisir dari tahun 2013 sampai dengan saat ini, tercatat terdapat 5.477.618 koleksi perpustakaan yang terdiri dari buku, majalah, jurnal, CD dan lain sebagainya. Sementara jumlah transaksi peminjaman setiap tahunnya terus berkembang, walaupun perkembangannya belum signifikan, tercatat sekitar 3% setiap harinya. Untuk membantu proses analisis pola data perpustakaan, telah dibuat sistem berbasis web dengan menggunakan teknologi OLAP (Online Analytical Process), dengan pemodelan data berbasis multidimensi data. Untuk membuktikan kesederhanaan perancangan data berbasis multidimensi, waktu akses dan efisiensi join, pada penelitian ini pemodelan data yang pilih adalah star schema. Hasil pengujian menunjukkan bahwa kinerja star skema pada data perpustakaan dipengaruhi oleh jumlah data, proses pembersihan data serta jumlah join.

Kata kunci: *star schema, OLAP, perpustakaan*

Abstract

Library information system whose data began to be organized from 2013 to the present, has recorded 5,477,618 library collections consisting of books, magazines, journals, CDs and so on. While the number of lending transactions continues to grow each year, although the development is not yet significant, it is recorded at around 3% every day. To help the process of analyzing library data patterns, a web-based system has been created using OLAP (Online Analytical Process) technology, with multidimensional data-based data modeling. To prove the simplicity of multidimensional data-based design, access time and join efficiency, in this study the selected data modeling is a star schema. The test results show that the star schema performance in library data is influenced by the amount of data, the cleaning process and the number of joins.

Keywords: *Star schema, OLAP, Lybrary*

1. Pendahuluan

Dalam sebuah sistem informasi data perpustakaan, terdapat beberapa bisnis proses yang dikelola, diantaranya proses pencatatan pengadaan buku, proses penataan buku ke dalam rak, proses peminjaman dan pengembalian, serta perlakuan denda. Demikian juga halnya dengan perpustakaan di Politeknik Caltex Riau yang datanya mulai terorganisir dari tahun 2013 sampai dengan saat ini, tercatat terdapat 5.477.618 koleksi perpustakaan yang terdiri dari buku, majalah, jurnal, CD dan lain sebagainya. Sementara jumlah transaksi peminjaman setiap tahunnya terus berkembang, walaupun perkembangannya belum signifikan. Berdasarkan pengamatan langsung terhadap data perpustakaan sampai dengan tahun 2017 tercatat peningkatan data terjadi sekitar 3% setiap harinya. Semua transaksi tersebut saat ini dikelola dalam sebuah system informasi perpustakaan yang beralamat di pustaka.pcr.ac.id. Namun berdasarkan hasil survey dan wawancara yang dilakukan kepada staf perpustakaan, terdapat beberapa kekurangan dari system yang ada saat ini, diantaranya adalah terbatasnya informasi yang bisa diperoleh oleh pengelola terutama informasi yang dapat membantu proses analisis, seperti pola peminjaman buku oleh mahasiswa dan dosen, koleksi buku yang banyak diminati atau sebaran koleksi perpustakaan yang banyak dibutuhkan.

Salah satu teknologi yang dapat menghasilkan informasi tersebut adalah teknologi OLAP (*Online Analytical Process*). Dengan menggunakan teknologi OLAP, informasi yang dihasilkan dapat melibatkan berbagai dimensi dan sudut pandang yang berbeda, dapat melakukan penelusuran data secara rinci sampai pada tingkat kesimpulan atau global, serta dapat membandingkan beberapa data dalam satu tampilan. *Star schema* merupakan skema pada pemodelan multidimensi yang paling sederhana, dimana pada setiap dimensinya tidak memiliki tingkatan atau tidak terdapat sub dimensi. Menurut Agung Prasetyo dalam penelitiannya (Agung Prasetyo, 2017), kinerja *query* pada skema bintang lebih cepat dibandingkan dengan pemodelan lainnya. Namun pada penelitian tersebut belum dilakukan penelitian factor apa saja yang mempengaruhi kinerja tersebut. Apakah jumlah data, jumlah dimensi, atau jumlah *measurement* mempengaruhi kinerja tersebut.

Sehingga tujuan dari penelitian ini diantaranya adalah :

1. Melakukan pemodelan data perpustakaan dengan *star schema*.
2. Menguji kinerja pemodelan *star schema*, dari segi kesederhanaan perancangan, volume penyimpanan, efisiensi join, query OLAP dan kinerja CRUD.
3. Melakukan operasi OLAP untuk menghasilkan informasi yang bermanfaat bagi pengelola perpustakaan.

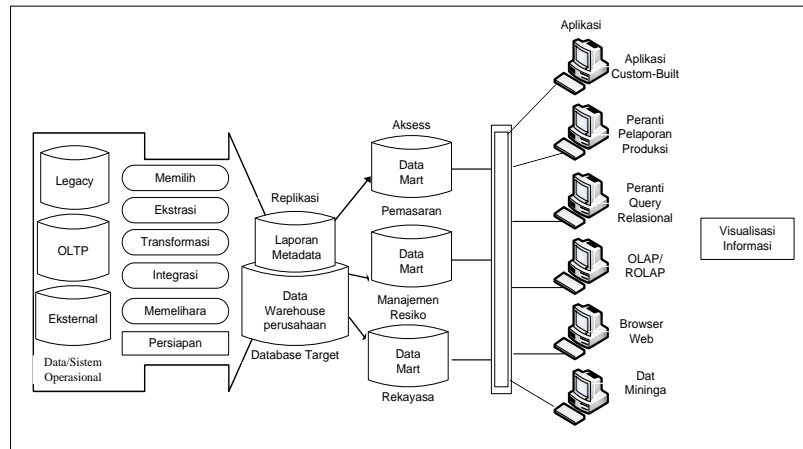
2. Tinjauan Pustaka

Beberapa studi literatur yang telah dilakukan penulis untuk mendukung penelitian ini adalah mempelajari mengenai alur transaksi di perpustakaan PCR, mempelajari mengenai data warehouse, OLAP, serta pembuatan dashboard berbasis web.

2.1 Data Warehouse

Data *warehouse* adalah koleksi data yang bersifat *subject-oriented*, terintegrasi, *time variant*, dan *non-volatile* yang digunakan untuk mendukung proses pengambilan keputusan yang strategis untuk perusahaan.

Data yang ada didalam data *warehouse* bisa berasal dari banyak sumber, misalkan dari basis data operasional atau transaksional dan sumber dari luar misalkan dari *web*, penyedia jasa informasi, dari perusahaan lain, dan sebagainya.



2.2 OLAP

OLAP merupakan salah satu data *warehouse tools* untuk melakukan analisis data. OLAP sendiri adalah suatu teknologi yang dirancang untuk memberikan kinerja yang unggul untuk ad hoc business *intelligence queries* (Ilmiah et al., 2016b). OLAP dirancang untuk beroperasi secara efisien dengan data yang terorganisir sesuai dengan model dimensi umum yang biasa digunakan dalam data warehouse. Beberapa operasi OLAP adalah :

- 1) *Consolidation (roll-up)*: Konsolidasi melibatkan pengelompokkan data untuk melihat data secara global atau rangkuman (*summary*).
- 2) *Drill-down*: Suatu bentuk yang merupakan kebalikan dari konsolidasi, untuk mendapatkan lebih detail
- 3) *Slicing and dicing*: *Slicing* dan *dicing* adalah operasi untuk melihat data sebagai visualisasi dari kubus
- 4) *Pivot* Menampilkan nilai-nilai ukuran dalam tata letak tabel yang berbeda dan juga bisa mengatur kembali dimensi dalam OLAP *cube*.

2.3 Nine-step Methodology

Metode perancangan *data warehouse* menurut Kimball yang digunakan meliputi 9 tahap yang dikenal dengan *nine-step Methodology*. Kesembilan tahap itu yaitu (Mulyati et al., 2014):

- 1) Memilih proses (*choosing the process*), yaitu menentukan proses bisnis yang akan masuk dalam ruang lingkup system.
- 2) Memilih grain (*choosing grain*), yaitu menentukan fakta-fakta yang akan menjadi subjek analisis dalam membangun data warehouse.
- 3) Mengidentifikasi dan penesuaian dimensi (*identifying and conforming the dimension*), yaitu melakukan identifikasi dimensi yang berhubungan dengan fakta yang telah ditentukan sebelumnya.
- 4) Memilih Fakta (*choosing the fact*), yaitu lebih spesifik menentukan fakta dari *grain* yang sudah ditentukan sebelumnya.
- 5) Menyimpan perhitungan table fakta (*storing pre-calculation in the fact table*), yaitu melakukan penyimpanan dan perhitungan awal *measurement* yang akan digunakan pada table fakta.

- 6) Melengkapi table dimensi (*rounding out the dimension table*), yaitu melengkapi semua table dimensi dengan atribut yang dibutuhkan beserta tipe data dan keterangan yang diperlukan.
- 7) Pemilihan durasi database (*choosing the duration of database*), yaitu penentuan masa penggunaan data.
- 8) Penentuan perubahan dimensi (*determine the need to track slowly changing dimension*), yaitu melakukan penetapan perubahan data yang berpengaruh terhadap perubahan dimensi.
- 9) Perancangan fisik data warehouse (*decide the physical design*), yaitu tahapan melakukan perancangan fisik data warehouse.

3. Metode Penelitian

Metode penelitian yang dilakukan pada penelitian ini, terbagi menjadi tiga tahapan yaitu:

- 1) Analisis dan Perencanaan, proses ini akan dilakukan dengan mengikuti *nine steps Kimball*, yang dimulai dengan melakukan analisis kebutuhan system, melakukan perancangan dengan pendekatan pemodelan multidimensi data dan akan menghasilkan skema multidimensi berupa star skema.
- 2) Pembangunan data warehouse. Tahapan dilakukan untuk mengimplemantasikan hasil dari perancangan data warehouse menggunakan perangkat lunak *Tableau* atau *Pentaho*. Proses implementasi ini dilakukan melalui tahapan ETL atau pembersihan data, pemodelan atau pemisahan antara dimensi dan measurement, serta visualisasi.
- 3) Pengujian. Sedangkan metode pengujian yang akan digunakan adalah pengujian kinerja dari star skema dengan cara melakukan uji performance query.

4. Perancangan

Data transaksi perpustakaan berupa data koleksi buku, data peminjaman, data pengembalian, serta data kunjungan perpustakaan saat ini masih berupa data semi terstruktur dalam ekstensi .xls. Data tersebut akan dijadikan sumber data pada penelitian ini. Adapun langkah-langkah yang dilakukan dalam pembangunan data warehouse ini mengacu pada metode *nine step kimbal*, sebagai berikut :

- 1) Memilih Proses
Proses dari perpustakaan PCR yang digunakan untuk membuat data *warehouse* adalah proses pengolahan data mahasiswa, pengolahan data peminjam dan pengolahan data pengunjung.
- 2) Memilih *Grain*
Setelah proses ditentukan, maka tahap selanjutnya adalah menentukan *grain*. Pemilihan *grain* berarti menentukan secara tepat apa yang akan dipresentasikan oleh *record* pada tabel fakta. *Grain* dari perpustakaan PCR yang digunakan dalam membuat data *warehouse* adalah:

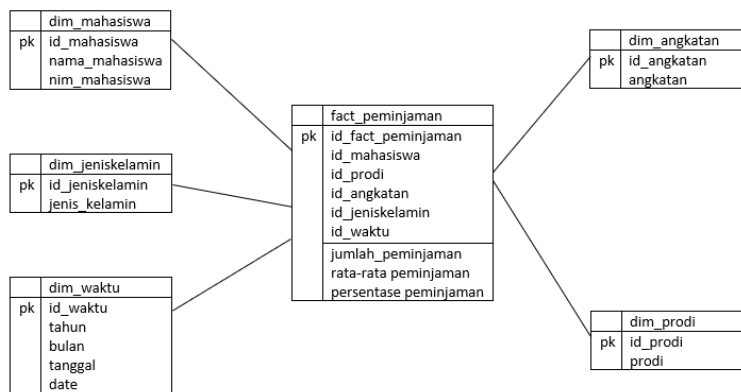
Tabel 3.1 *Grain*

| Grain | Data Source |
|--|-------------------------------|
| Mengetahui pola peminjam pada perpustakaan | <i>Database dan Flat File</i> |
| Mengetahui partisipasi peminjaman dan | <i>Database dan Flat</i> |

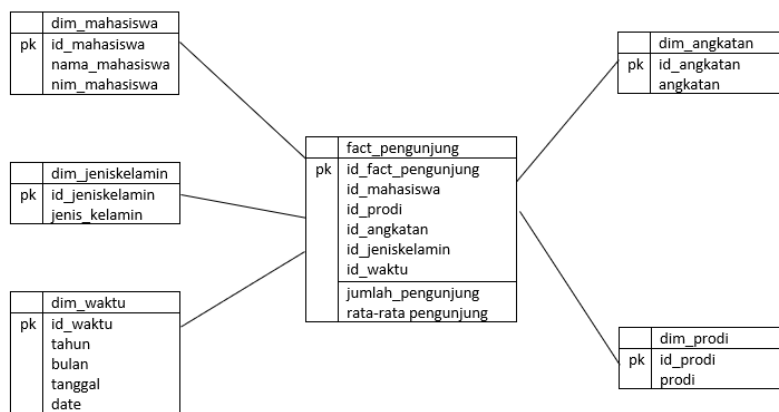
| | |
|-------------------------|-------------|
| pengunjung perpustakaan | <i>File</i> |
|-------------------------|-------------|

- 3) Mengidentifikasi dan Penyesuaian Dimensi
Langkah ketiga dalam membuat *data warehouse* adalah mengidentifikasi dimensi yang berhubungan dengan tabel fakta. Dari identifikasi maka dapat ditentukan dimensi-dimensi yang terlibat. Dari tabel fakta peminjaman dimensi yang terlibat yaitu dimensi mahasiswa, dimensi prodi, dimensi angkatan, dimensi jenis kelamin dan dimensi waktu, sedangkan pada tabel fakta pengunjung dimensi yang terlibat yaitu dimensi mahasiswa, dimensi prodi, dimensi angkatan, dimensi jenis kelamin dan dimensi waktu.

- 4) Memilih Fakta
Langkah keempat dalam membuat *data warehouse* adalah memilih tabel fakta berdasarkan *grain* sebelumnya. Tabel fakta ini berisikan atribut-atribut kunci yang berasal dari tabel dimensi yang telah dibuat pada tahap sebelumnya. Permodelan yang digunakan adalah model *star schema*. Berikut merupakan gambar tabel fakta yang akan dibuat.



Gambar 3.1 Perancangan *Star Schema* pada Tabel Fakta Peminjaman



Gambar 3.2 Perancangan *Star Schema* pada Tabel Fakta Pengunjung

- 5) Menyimpan Pre-calculation pada Tabel Fakta
Langkah kelima dalam membuat *data warehouse* adalah menyimpan perhitungan awal pada tabel fakta. Melengkapi Tabel Dimensi

- 6) Langkah ke enam dalam membuat *data warehouse* adalah melengkapi tabel dimensi. Kelengkapan tabel dimensi berisi data antara lain nama field, tipe data dan panjang data. Tabel dimensi beserta atributnya antara lain sebagai berikut:

Tabel 3.2 Dimensi Mahasiswa

| Field | Tipe dan panjang data | Keterangan |
|----------------|------------------------------|-------------------|
| id_mahasiswa | int (50) | Id mahasiswa |
| nama_mahasiswa | varchar (255) | Nama mahasiswa |
| nim_mahasiswa | int (50) | Nim mahasiswa |

Tabel 3.3 Dimensi Angkatan

| Field | Tipe dan panjang data | Keterangan |
|--------------|------------------------------|-------------------|
| id_angkatan | int (50) | Id angkatan |
| angkatan | int (255) | Angkatan |

Tabel 3.4 Dimensi Jenis Kelamin

| Field | Tipe dan panjang data | Keterangan |
|-----------------|------------------------------|-------------------|
| id_jeniskelamin | int (50) | Id jenis kelamin |
| jenis_kelamin | varchar (255) | Jenis kelamin |

Tabel 3.5 Dimensi Prodi

| Field | Tipe dan panjang data | Keterangan |
|--------------|------------------------------|-------------------|
| id_prodi | int (50) | Id prodi |
| prodi | varchar (255) | prodi |

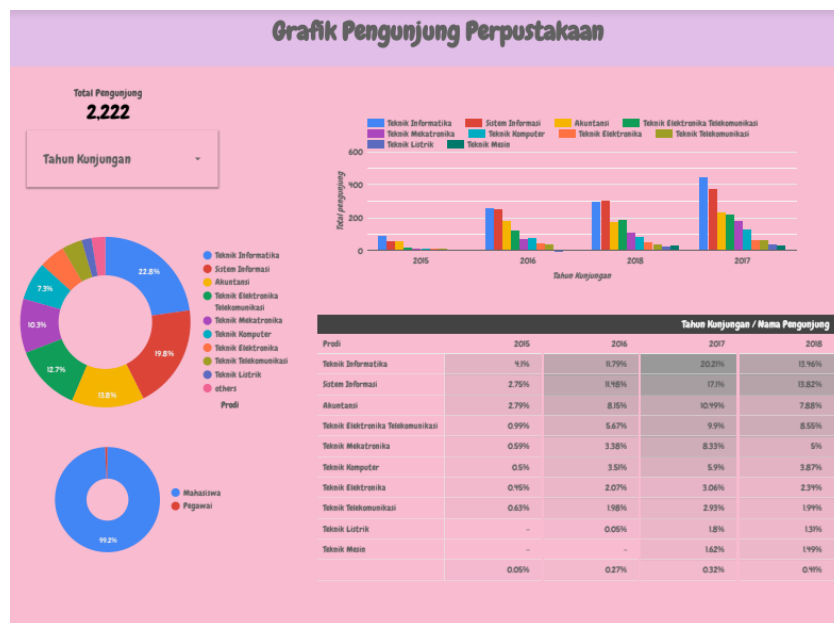
Tabel 3.6 Dimensi Waktu

| Field | Tipe dan panjang data | Keterangan |
|--------------|------------------------------|-------------------|
|--------------|------------------------------|-------------------|

| | | |
|----------|----------|----------|
| id_waktu | Int (50) | Id waktu |
| tahun | Int (50) | Tahun |
| bulan | Int (50) | Bulan |
| tanggal | Int (50) | Tanggal |
| date | Date | Date |

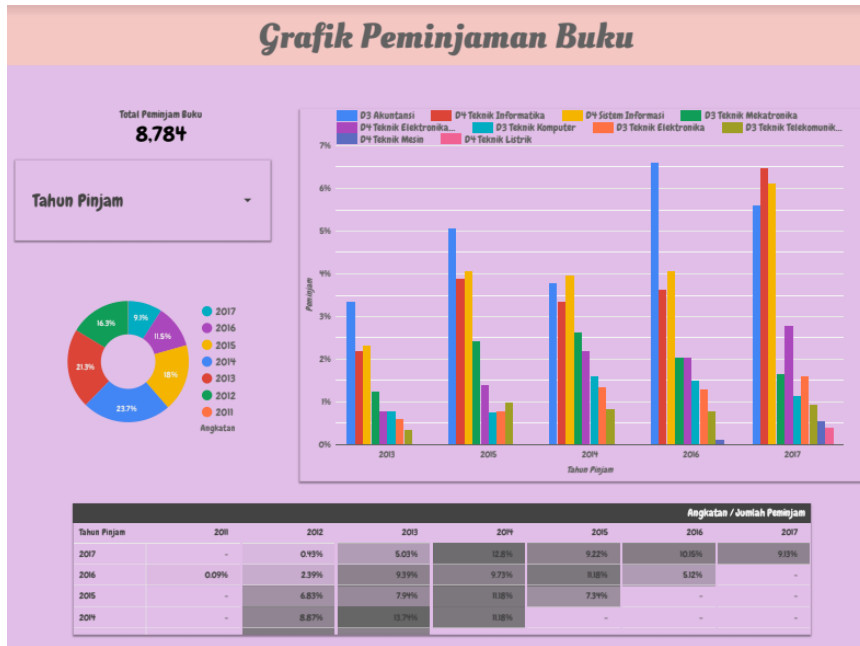
- 7) Pemilihan Durasi Basis Data
- 8) Menelusuri Perubahan Dimensi
- 9) Penentuan Prioritas *Model Query* dan Memilih *Physical Design*

5. Hasil



Gambar 3 Dashboard Penunjang Perpustakaan

Pada dashboard ini terdapat beberapa grafik informasi dengan beberapa jenis grafik yaitu grafik pie untuk melihat informasi sebaran pengunjung berdasarkan prodi, grafik batang untuk menampilkan informasi sebaran pengunjung berdasarkan tahun, serta table untuk rincian tahun dan prodi pengunjung.



Gambar 4 Dashboard Peminjaman Buku

1.3.2 Pengujian Kinerja Skema Bintang

Tahap pengujian ini dilakukan untuk melihat kinerja dari *star schema* pada saat melakukan proses OLAP. Kinerja yang akan diuji terkait waktu eksekusi dari proses *filter* data pada halaman laporan peminjam dan laporan peminjaman dan pengunjung. Berikut merupakan potongan *coding* untuk proses *join* pada tampilan halaman peminjaman.

```

270 <?php
271 if (isset($_GET['tahun']) || isset($_GET['angkatan']) || isset($_GET['prodi']) || isset($_GET['jenis_kelamin'])) {
272     $data=$this->db->query("SELECT COUNT(fact_peminjaman.id_fact_peminjaman) jumlah , dim_jeniskelamin.jenis_kelamin
273 FROM `fact_peminjaman` , dim_jeniskelamin,dim_prodi,dim_angkatan,dim_waktu
274 WHERE dim_jeniskelamin.id_jeniskelamin=fact_peminjaman.id_jeniskelamin
275 and dim_prodi.id_prodi=fact_peminjaman.id_prodi
276 and dim_angkatan.id_angkatan=fact_peminjaman.id_angkatan
277 and dim_waktu.id_waktu=fact_peminjaman.id_waktu ". $query ." ". $query_tahun ." ". $query_angkatan ." ". $query_prodi ."
278 GROUP BY dim_jeniskelamin.jenis_kelamin")->result();
279 }else{
280     $data=$this->db->query("SELECT COUNT(fact_peminjaman.id_fact_peminjaman) jumlah , dim_jeniskelamin.jenis_kelamin
281 FROM `fact_peminjaman` , dim_jeniskelamin,dim_prodi,dim_angkatan,dim_waktu
282 WHERE dim_jeniskelamin.id_jeniskelamin=fact_peminjaman.id_jeniskelamin
283 and dim_prodi.id_prodi=fact_peminjaman.id_prodi
284 and dim_angkatan.id_angkatan=fact_peminjaman.id_angkatan
285 and dim_waktu.id_waktu=fact_peminjaman.id_waktu
286 GROUP BY dim_jeniskelamin.jenis_kelamin")->result();
287 }
288
289 for ($i=0; $i < count($data); $i++) {
290     echo "<br>";
291     name: "" $data[$i]->jenis_kelamin."" ,
292     y: "" $data[$i]->jumlah."" ,
293     sliced: true,
294     selected: true
295     },"";
296 }
297 >>
    
```

Gambar 4.5 Potongan Coding Proses Join Halaman Peminjaman

Potongan *coding* diatas menjelaskan bahwa pada halaman peminjaman memiliki 4 dimensi yang saling *join* yaitu dimensi jenis kelamin, prodi, angkatan dan waktu. Berdasarkan pengujian laporan peminjam di atas, diperoleh waktu dari proses eksekusi setiap filter yang dapat dilihat pada tabel dibawah ini.

Tabel 4.1 Pengujian Kinerja pada Fact Peminjaman dan Fact Pengunjung

| Tabel Fakta | Dimensi Yang Terlibat | Jumlah Data | Jumlah | Waktu Yang Diharapkan | Waktu |
|-------------|-----------------------|-------------|--------|-----------------------|-------|
|-------------|-----------------------|-------------|--------|-----------------------|-------|

| | | | Join | | Sebenarnya |
|-----------------|--|-------|------|--------------|------------|
| Fact Peminjaman | <ul style="list-style-type: none"> - Dimensi Mahasiswa - Dimensi Angkatan - Dimensi Jenis Kelamin - Dimensi Prodi - Dimensi Waktu | 3.000 | | 2 - 12 detik | 0.42 detik |
| | | 6.000 | | 2 - 12 detik | 0.55 detik |
| | | 8.785 | | 2 - 12 detik | 0.60 detik |

```

208 <?php
209
210 if (isset($_GET['tahun'])||isset($_GET['angkatan'])||isset($_GET['prodi'])||isset($_GET['jenis_kelamin'])) {
211     $data_rataPeminjaman=$this->db->query("SELECT count(fact_peminjaman.id_fact_peminjaman)/5 jumlah, dim_waktu.tahun
212     FROM fact_peminjaman , dim_jenis_kelamin,dim_prodi,dim_angkatan,dim_waktu
213     WHERE dim_jenis_kelamin.id_jenis_kelamin=fact_peminjaman.id_jenis_kelamin
214     and dim_prodi.id_prodi=fact_peminjaman.id_prodi
215     and dim_angkatan.id_angkatan=fact_peminjaman.id_angkatan
216     and dim_waktu.id_waktu=fact_peminjaman.id_mahasiswa ".$query." ".$query_tahun." ".$query_angkatan." ".$query_prodi."
217     GROUP BY dim_waktu.tahun")->result();
218 }else{
219     $data_rataPeminjaman=$this->db->query("SELECT count(fact_peminjaman.id_fact_peminjaman)/5 jumlah, dim_waktu.tahun
220     FROM fact_peminjaman , dim_jenis_kelamin,dim_prodi,dim_angkatan,dim_waktu
221     WHERE dim_jenis_kelamin.id_jenis_kelamin=fact_peminjaman.id_jenis_kelamin
222     and dim_prodi.id_prodi=fact_peminjaman.id_prodi
223     and dim_angkatan.id_angkatan=fact_peminjaman.id_angkatan
224     and dim_waktu.id_waktu=fact_peminjaman.id_mahasiswa ".$query." ".$query_tahun." ".$query_angkatan." ".$query_prodi."
225     GROUP BY dim_waktu.tahun")->result();
226 } ??
227
228 <?php
229
230 if (isset($_GET['tahun'])||isset($_GET['angkatan'])||isset($_GET['prodi'])||isset($_GET['jenis_kelamin'])) {
231     $data_ratakunjungan=$this->db->query("SELECT count(fact_pengunjung.id_fact_pengunjung)/5 jumlah, dim_waktu.tahun
232     FROM fact_pengunjung , dim_jenis_kelamin,dim_prodi,dim_angkatan,dim_waktu
233     WHERE dim_jenis_kelamin.id_jenis_kelamin=fact_pengunjung.id_jenis_kelamin
234     and dim_prodi.id_prodi=fact_pengunjung.id_prodi
235     and dim_angkatan.id_angkatan=fact_pengunjung.id_angkatan
236     and dim_waktu.id_waktu=fact_pengunjung.id_mahasiswa ".$query." ".$query_tahun." ".$query_angkatan." ".$query_prodi."
237     GROUP BY dim_waktu.tahun")->result();
238 }else{
239     $data_ratakunjungan=$this->db->query("SELECT count(fact_pengunjung.id_fact_pengunjung)/5 jumlah, dim_waktu.tahun
240     FROM fact_pengunjung , dim_jenis_kelamin,dim_prodi,dim_angkatan,dim_waktu
241     WHERE dim_jenis_kelamin.id_jenis_kelamin=fact_pengunjung.id_jenis_kelamin
242     and dim_prodi.id_prodi=fact_pengunjung.id_prodi
243     and dim_angkatan.id_angkatan=fact_pengunjung.id_angkatan
244     and dim_waktu.id_waktu=fact_pengunjung.id_mahasiswa ".$query." ".$query_tahun." ".$query_angkatan." ".$query_prodi."
245     GROUP BY dim_waktu.tahun")->result();
246 } ??
    
```

Gambar 4.6 Potongan Coding Proses Join Halaman Peminjam dan Pengunjung

Potongan coding diatas terdapat query untuk peminjam dan query untuk pengunjung. Query peminjam memiliki 4 dimensi yang saling join yaitu dimensi jenis kelamin, prodi, angkatan, dan waktu, dan untuk query pengunjung memiliki 4 dimensi yang saling join yaitu dimensi jenis kelamin, prodi, angkatan dan waktu. Berdasarkan pengujian laporan peminjam di atas, diperoleh waktu dari proses eksekusi setiap filter yang dapat dilihat pada tabel dibawah ini.

Tabel 4.2 Pengujian Kinerja pada Fact Peminjaman dan Fact Pengunjung

| Tabel Fakta | Dimensi Yang Terlibat | Jumlah Data | Jumlah Join | Waktu Yang Diharapkan | Waktu Sebenarnya |
|-------------------------------------|--|-------------|-------------|-----------------------|------------------|
| Fact Peminjaman dan Fact Pengunjung | <ul style="list-style-type: none"> - Dimensi Mahasiswa - Dimensi Angkatan | 9.738 | 8 | 2 - 12 detik | 0.45 detik |
| | <ul style="list-style-type: none"> - Dimensi Jenis Kelamin - Dimensi Prodi | 19.476 | | 2 - 12 detik | 0.55 detik |
| | <ul style="list-style-type: none"> - Dimensi Waktu | 29.214 | | 2 - 12 detik | 0.60 detik |

Berdasarkan hasil yang telah dijabarkan diatas, lama waktu eksekusi yang dilakukan pada halaman laporan peminjam dan laporan peminjaman dan pengunjung dengan menggunakan pemodelan *star schema* yaitu tidak lebih dari 3 detik, sehingga masuk kedalam kategori cepat atau tanpa delay. Maka dapat disimpulkan bahwa pemodelan *data warehouse* dengan menggunakan *star schema* akan lebih mempersingkat proses eksekusi saat memvisualisasikan informasi yang diperoleh dari berbagai sumber yang berbeda.

6. Kesimpulan dan Saran

6.1 Kesimpulan

Setelah dilakukan pengujian terhadap sistem yang telah dibangun, didapatkan kesimpulan sebagai berikut:

- 1) Dengan membuat sistem yang menerapkan metode nine-step kimball dengan pemodelan *star schema* dapat menghasilkan informasi berupa pola peminjam dan pengunjung pada perpustakaan PCR.
- 2) Pemodelan *star schema* pada data mahasiswa, data peminjaman dan data pengunjung melibatkan *join* antar dimensi yang berpengaruh pada kecepatan waktu eksekusi. Beberapa hal yang mempengaruhi waktu eksekusi adalah jumlah data, kualitas data dan jumlah *join*.

6.2 Saran

Saran untuk penelitian lanjutan dari penelitian ini adalah melakukan pemodelan lain selain dari pemodelan bintang, serta meningkatkan proses ETL untuk meningkatkan kualitas data

Daftar Pustaka

- [1] Ardanu, F., Himawan, H., & P, D. B. (2013). PEMANFAATAN TEKNOLOGI DATA MINING DALAM MENENTUKAN EFEKTIFITAS PENYEBARAN BROSUR.
- [2] Dharmayanti, D., Bachtiar, A. M., & Heryandi, A. (2013). Pemodelan Data Warehouse, *12*(2), 151–168.
- [3] Fadilah, U., Winarno, W. W., Amborowati, A., Fadilah, U., Winarno, W. W., & Amborowati, A. (2016). Perancangan Data Warehouse Untuk Sistem Akademik STMIK Kadiri Data Warehouse System Design for Academic STMIK Kadiri, *6*(2), 217–228.
- [4] Ilmiah, J., Komputa, I., Volume, E., Issn, F., Cv, D. I., Anugerah, K., ... Bandung, U. (2016a). PEMBANGUNAN PERANGKAT LUNAK DATA WAREHOUSE Jurnal Ilmiah Komputer dan Informatika (KOMPUTA), *1*.
- [5] Ilmiah, J., Komputa, I., Volume, E., Issn, F., Cv, D. I., Anugerah, K., ... Bandung, U. (2016b). PEMBANGUNAN PERANGKAT LUNAK DATA WAREHOUSE Jurnal Ilmiah Komputer dan Informatika (KOMPUTA). ok
- [6] Mulyati, S., Amini, S., & Juliasari, N. (2014). 104-279-1-PB.pdf. *Jurnal Telematika MKOM*, *6 No.1*.