

Multi Join Query Pada Proyek Sistem Monitoring dan Evaluasi Pembangunan

Saputro Andi Pamungkas

Jurusan Teknik Informatika, Fakultas Teknologi Industri
Universitas Islam Indonesia
Yogyakarta
16523189@students.uii.ac.id

Hanson Prihantoro Putro

Jurusan Teknik Informatika, Fakultas Teknologi Industri
Universitas Islam Indonesia
Yogyakarta
hanson@uii.ac.id

Abstrak—Sistem Monitoring dan Evaluasi Pembangunan atau biasa disingkat dengan SMEP merupakan website pelaporan realisasi anggaran Satuan Kerja Perangkat Daerah. Banyaknya pengolahan yang ada pada sistem SMEP membuat hasilnya tidak efisien, seperti query yang lambat atau data yang diambil salah. Query yang efisien dapat membantu *stakeholder* pengguna SMEP, kerjanya bisa menjadi lebih cepat, mudah untuk membaca data serta dapat memudahkan dalam mengambil keputusan. Maka dari itu, bagaimana cara melakukan *query* menjadi penting, salah satu caranya adalah dengan Multi Join Query untuk mendapatkan *query* data yang berjenjang. Pada makalah ini akan disajikan bagaimana melakukan Multi Join Query, sehingga harapannya dapat menjadi pembelajaran tentang bagaimana membuat *query* yang efisien.

Keywords—Sistem Monitoring dan Evaluasi Pembangunan; Multi Join Query.

I. PENDAHULUAN

Sejak dilangsungkannya otonomi daerah, maka saat ini daerah memiliki keleluasaan dalam mengatur segala urusan pembangunan daerahnya secara mandiri. Daerah selain bertugas untuk memberikan pelayanan kepada masyarakat dituntut serta untuk melaksanakan pembangunan secara efektif dan efisien. Salah satunya adalah dalam penyerapan anggaran pembangunan. Saat ini Bagian Administrasi Pembangunan Kota Probolinggo memiliki 39 Satuan Kerja Perangkat Daerah atau biasa di singkat dengan SKPD [1]. Masing-masing SKPD memiliki program dan kegiatan sendiri. Sesuai dengan tuntutan bahwa pemerintah daerah perlu efisien maka daerah dituntut untuk melakukan proses monitoring dan evaluasi dari penyerapan anggaran oleh masing-masing SKPD. Untuk itu Pemerintah Kota Probolinggo membuat website SMEP Kota Probolinggo. Sistem ini nantinya yang akan digunakan untuk memasukkan realisasi anggaran dan capaian yang sudah dicapai oleh masing-masing SKPD. Dengan adanya Sistem Monitoring dan Evaluasi Pembangunan (SMEP) maka dapat diketahui SKPD mana saja yang hasil kerjanya baik dan begitu pula sebaliknya.

Website SMEP menggunakan DBMS MySQL untuk penyimpanan data. Data yang ada akan diolah dengan beberapa tabel lainnya dengan cara melakukan *join table*. Karena data dalam database SMEP memiliki banyak tabel terkait, maka dibutuhkan *query* dari beberapa tabel atau Multi Join Query

untuk melakukan pengolahan data. Harapannya setelah makalah ini selesai, dapat memberi pengetahuan bagi para praktisi untuk membuat *query* yang efisien.

II. KAJIAN PUSTAKA

Sistem Monitoring dan Evaluasi Pembangunan merupakan suatu sistem yang berfungsi untuk menghimpun, menganalisis dan menyusun laporan hasil evaluasi dan monitoring realisasi kegiatan SKPD[2]. SKPD sebagai pelaksana fungsi eksekutif di daerah memiliki tanggung jawab untuk menjalankan rancangan program atau kegiatan. Setelah itu, masing-masing SKPD wajib melaporkan realisasi dari program atau kegiatan yang sudah dijalankan. Laporan tersebut dapat dimasukkan kedalam sistem SMEP.

Join statement adalah sebuah *select statment* dengan lebih dari satu table dengan klausa *from*[3]. Dua buah table dapat disatukan dengan cara melakukan *join* antara kolom *primary key* dengan kolom *foreign key*. *Join* juga dapat ditambahkan dengan kondisi tertentu, bisa dengan menggunakan klausa *where*. *Join* tabel terdiri dari beberapa ada *inner join*, *full outer join*, *union join*, *left join* dan *right join*. Pemilihan tipe *join* berpengaruh kepada hasil *query* yang didapatkan. Ada saatnya hasil yang ingin kita dapatkan membutuhkan *join* dari banyak table. Untuk kasus seperti itu dinamakan *Multi Join Query*. *Multi Join Query* adalah satu *query* yang memiliki lebih dari satu *join*[4]. Penggunaan *Multi Join Query* hampir sama dengan *join* tabel pada umumnya, hal yang membedakan adalah jumlah tabel yang akan dilakukan *join*.

Salah satu pembahasan mengenai *multi join query* berupa optimasi pada *query*. *Multi join query* sangat krusial yang dapat berakibat pada kapabilitas database untuk mengolah data[5].

III. METODOLOGI

Metodologi yang digunakan berawal dari kebutuhan sistem sampai dengan implementasi *query* yang terdiri dari empat tahapan. Tahap pertama adalah ketika ada suatu *task* yang diberikan kepada programmer. Kemudian programmer menganalisis data dari *task* yang diberikan, kira-kira data apa saja yang perlu diolah atau ditampilkan oleh sistem. Setelah tahu apa saja data yang diperlukan, tahap selanjutnya adalah analisis tabel yang diperlukan beserta dengan tabel terkait lainnya untuk menampilkan data yang diharapkan. Tahap selanjutnya adalah perlu dibuat sebuah skema tentang cara

pengambilan data. Misalkan ada *join tabel* apakah harus menggunakan *left join*, *right join* atau *inner join*, menentukan prioritas tabel ketika *join*, menentukan *primary key* dan *foreign key* terkait, penggunaan *aggregate function*, menentukan apakah perlu menggunakan *having* atau *group by* dan yang terakhir apakah perlu menambahkan kondisional *where*, *in*, *if*, *case* untuk kasus tertentu. Langkah terakhir adalah implementasi query MySQL melalui *SQL client*. Tidak lupa juga selalu dilakukan pengujian apakah query yang sudah dibuat berjalan dengan optimal dengan tidak membutuhkan waktu yang lama.

IV. HASIL

A. Kebutuhan Sistem dan Analisis Data

Sistem ini mempunyai beberapa fitur diantaranya adalah Laporan Bulanan dan Dashboard. Fitur dashboard digunakan untuk menampilkan data capaian serapan anggaran SKPD dalam kurun waktu satu tahun. Sehingga Bagian Administrasi Pembangunan atau SKPD dapat mengetahui sudah berapa banyak anggaran yang sudah diserap beserta dengan SKPD yang memiliki capaian terbaik. Fitur selanjutnya adalah Laporan bulanan dimana merupakan sebuah fitur pencarian sejauh mana anggaran yang ada dapat diserap oleh SKPD. Nantinya laporan ini akan dicetak kemudian akan disahkan dengan cap basah.

Secara rinci data yang akan ditampilkan di kedua sistem tersebut banyak, terkhusus untuk laporan bulanan yang menyajikan tabel rekapitulasi keuangan perbulan beserta dengan grafik realisasi keuangan dan realisasi fisik. Sementara dashboard menampilkan jumlah anggaran dan realisasinya dalam kurun waktu satu tahun. Lebih lengkapnya apa saja yang ditampilkan ada dalam Tabel I.

TABLE I. ANALISIS KEBUTUHAN DATA

Fitur	Query	Data yang ditampilkan
Dashboard	SKPD Serapan Tertinggi	Nama SKPD, Realisasi Fisik, Prosentase
	SKPD Serapan Terendah	Nama SKPD, Realisasi Fisik, Prosentase
	Total Anggaran	Total Anggaran
	Total Serapan	Total Anggaran
	Sisa Anggaran	Sisa Anggaran
Laporan Bulanan	Grafik Target & Realisasi Keuangan	Target Keuangan, Realisasi Keuangan, Nama Bulan, Anggaran
	Grafik Realisasi Fisik	Realisasi Fisik, Nama Bulan, Anggaran

B. Analisis Tabel

Setelah mendapatkan data apa saja yang diperlukan untuk ditampilkan, maka langkah selanjutnya adalah melakukan analisis data tersebut diambil dari tabel apa. di bawah ini dijabarkan dari Tabel II apa saja data tersebut didapatkan.

TABLE II. QUERY TABEL

Query	Tabel
SKPD Serapan Tertinggi	smep_realisasi, dpa_dpa, mata_anggaran, rincian_mata_anggaran
SKPD Serapan Terendah	smep_realisasi, dpa_dpa, mata_anggaran, rincian_mata_anggaran
Total Anggaran	dpa_dpa, dpa_mata_anggaran
Total Serapan	dpa_dpa, dpa_mata_anggaran
Sisa Anggaran	dpa_dpa, dpa_mata_anggaran
Grafik Target & Realisasi Keuangan	dpa_mata_anggaran, dpa_dap, dpa_renc_pencairan, smep_realisasi
Grafik Realisasi Fisik	dpa_mata_anggaran, dpa_dap, dpa_renc_pencairan, smep_realisasi

Dari tabel diatas dapat disimpulkan bahwa terdapat sekitar lima tabel. Beberapa tabel tersebut saling terkait, tabel dpa_dpa memiliki mata anggaran yang berelasi dengan dpa_mata_anggaran, kemudian dpa_mata_anggaran mempunyai rincian yaitu dpa_rincian_mata_anggaran. Smep_realisasi merupakan tabel yang berisi input realisasi keuangan dari dpa, sehingga memiliki relasi dengan dpa_dpa.

C. Penyusunan Query dan Implementasi

Langkah selanjutnya adalah menyusun *query*, sebelum itu skema query harus dipetakan. Hal apa saja yang perlu dilakukan untuk mendapatkan *query* yang semestinya. Sepertinya yang tercantum dalam Tabel III.

TABLE III. SKEMA QUERY

Query	Skema Query
SKPD Serapan	Sum, Inner Join

Tertinggi	
SKPD Serapan Terendah	Sum, Inner Join
Total Anggaran	Sum, Inner Join
Total Serapan	Sum
Sisa Anggaran	Sum, Inner Join, Subquery
Grafik Target & Realisasi Keuangan	Sum, Inner Join, Where In
Grafik Realisasi Fisik	Sum, Inner Join, Where In

Berdasarkan hasil dari tabel diatas hampir semua *query* membutuhkan *join* dengan tabel lain. Selain itu, *sum* juga diperlukan karena semua data yang ditampilkan adalah hasil dari perhitungan.

Langkah selanjutnya adalah implementasi *query* yang sudah dibuat skemanya.

Query SKDP Serapan Terendah

```
SELECT @realisasi_fisik = SUM(realisasi_fisik)
FROM dpa_dpa
INNER JOIN smep_realisasi ON smep_realisasi.dpa_dpa_id =
dpa_dpa.id_dpa_dpa
WHERE tahun = now() AND
dpa_dpa.sikd_satker_id=sikd_satker.id_sikd_satker)
```

```
SELECT @volume = SUM(volume)
FROM dpa_mata_anggaran
INNER JOIN dpa_dpa ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran.dpa_dpa.id
WHERE dpa_dpa.sikd_satker_id = sikd_satker.id_sik_satker
```

```
SELECT @vol_rincian_ang = SUM(dpa_rincian_mata_anggaran.volume)
FROM dpa_rincian_mata_anggaran
INNER JOIN dpa_mata_anggaran ON
dpa_mata_anggaran.id_dpa_mata_anggaran=dpa_rincian_
mata_anggaran.dpa_mata_anggaran_id
INNER JOIN dpa_dpa ON
dpa_dpa.id_dpa_dpa=dpa_mata_anggaran.dpa_dpa_id
WHERE dpa_dpa.sikd_satker_id=sikd_satker.id_sikd_satker
```

```
SELECT sikd_satker.id_sikd_satker, sikd_satker.nama,
@realisasi_fisik as relaisasi_fisik,
@volume as volume_anggaran,
((@realisasi_fisik)/(@volume+@vol_rincian_angaran) * 100) as presentase
```

5 SKPD Terendah

SKPD SERAPAN TERENDAH	FISIK	PROSENTASE
DINAS LINGKUNGAN HIDUP	14666	0.95 %
DINAS PERUMAHAN DAN KAWASAN PERMUKIMAN	8090	1.20 %
DINAS PEMBERDAYAAN PEREMPUAN, PERLINDUNGAN ANAK DAN KELUARGA BERENCANA	20397	2.31 %
DINAS KESEHATAN	155669	4.06 %
DINAS KOMUNIKASI DAN INFORMATIKA	61200	10.74 %

Gambar 1. SKPD Terendah

Query SKDP Serapan Tertinggi

```
SELECT @realisasi_fisik = Sum(realisasi_fisik)
FROM dpa_dpa
INNER JOIN smep_realisasi ON smep_realisasi.dpa_dpa_id =
dpa_dpa.id_dpa_dpa
WHERE tahun = now() AND
dpa_dpa.sikd_satker_id=sikd_satker.id_sikd_satker)
```

```
SELECT @volume = SUM(volume)
FROM dpa_mata_anggaran
INNER JOIN dpa_dpa ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran.dpa_dpa.id
WHERE dpa_dpa.sikd_satker_id = sikd_satker.id_sik_satker
```

```
SELECT @vol_rincian_ang = SUM(dpa_rincian_mata_anggaran.volume)
FROM dpa_rincian_mata_anggaran
INNER JOIN dpa_mata_anggaran ON
dpa_mata_anggaran.id_dpa_mata_anggaran=dpa_rincian_
mata_anggaran.dpa_mata_anggaran_id
INNER JOIN dpa_dpa ON
dpa_dpa.id_dpa_dpa=dpa_mata_anggaran.dpa_dpa_id
WHERE dpa_dpa.sikd_satker_id=sikd_satker.id_sikd_satker
```

```
SELECT sikd_satker.id_sikd_satker,
sikd_satker.nama,
@realisasi_fisik as relaisasi_fisik,
@volume as volume_anggaran,
((@realisasi_fisik)/(@volume+@vol_rincian_angaran) * 100) as presentase
```

5 SKPD Tertinggi

SKPD SERAPAN TERTINGGI	FISIK	PROSENTASE
BADAN PENANGGULANGAN BENCANA DAERAH	88480	28.98 %
DINAS PERTANIAN DAN KETAHANAN PANGAN	201000.729999999998	27.21 %
INSPEKTORAT	184534	25.83 %
DINAS PENANAMAN MODAL DAN PELAYANAN TERPADU SATU PINTU	139168	23.91 %
DINAS PERIKANAN	571852	23.35 %

Gambar 2. SKPD Tertinggi

Total Anggaran

```
SELECT SUM(jumlah)
FROM dpa_dpa
INNER JOIN dpa_mata_anggaran ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran
WHERE status_dpa = 1 AND jns_dpa = DpaSkpdKgtn
```

TOTAL ANGGARAN

RP. 818.751.187.964

Gambar 3. Total Anggaran

Total Serapan

```
SELECT SUM(realisasi_keuangan)
FROM smep_realisasi
WHERE tahun = now()
```

TOTAL SERAPAN

RP. 174.918.775.491

Gambar 4. Total Serapan Anggaran

Sisa Anggaran

```
SELECT SUM(jumlah) - (
SELECT SUM(realisasi_keuangan) FROM smep_realisasi WHERE
tahun = now()
) as sisa_anggaran
FROM dpa_dpa
INNER JOIN dpa_mata_anggaran ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran.dpa_dpa.id
WHERE status_dpa = 1 AND
jns_dpa = 'DpaSkpdKgtn'
```

SISA ANGGARAN

RP. 643.832.412.472

Gambar 5. Sisa Anggaran

Grafik Target & Realisasi Keuangan

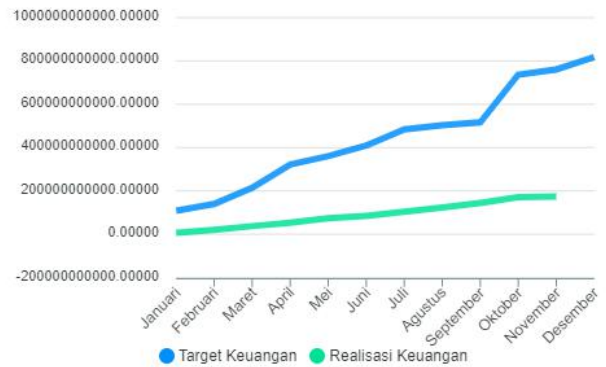
Langkah pertama adalah iterasi untuk mendapatkan realisasi keuangan setiap bulannya.

```
SELECT @realisasi = SUM(realisasi_keuangan)
FROM smep_realisasi
WHERE tahun = now() AND
bulan IN (1,2,3,4,5,6,7,8,9,10,11,12)
```

Setelah itu melakukan perhitungan untuk target keuangan dengan cara melakukan penjumlahan pada setiap iterasinya.

```
SELECT CONCAT(SUM(dpa_mata_anggaran.volume),?), @anggaran =
SUM(dpa_mata_anggaran.volume) as anggaran
FROM dpa_mata_anggaran
INNER JOIN dpa_dpa ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran.dpa_dpa.id
INNER JOIN dpa_renc_pencairan ON
dpa_renc_pencairan.dpa_mata_anggaran_id =
dpa_mata_anggaran.id_dpa_mata_anggaran
WHERE dpa_dpa.status_dpa = 1 AND
dpa_dpa.jns_dpa = 'DpaSkpdKgtn'
```

Grafik Target & Realisasi Keuangan



Gambar 6. Grafik Target & Realisasi Keuangan

Grafik Realisasi Fisik

```
SELECT @anggaran = SUM(dpa_mata_anggaran.volume)
FROM dpa_mata_anggaran
INNER JOIN dpa_dpa ON dpa_dpa.id_dpa_dpa =
dpa_mata_anggaran.dpa_dpa.id
WHERE dpa_dpa.status_dpa = 1 AND
dpa_dpa.jns_dpa = 'DpaSkpdKgtn'
```

```
SELECT @rincian = SUM(dpa_rincian_mata_anggaran.volume)
FROM dpa_rincian_mata_anggaran
INNER JOIN dpa_dpa ON dpa_dpa.id_dpa_dpa =
dpa_rincian_mata_anggaran.dpa_dpa.id
INNER JOIN dpa_mata_anggaran ON
dpa_rincian_mata_anggaran.dpa_mata_anggaran_id =
dpa_mata_anggaran.id_dpa_mata_anggaran
WHERE dpa_dpa.status_dpa = 1 AND
dpa_dpa.jns_dpa = 'DpaSkpdKgtn' AND
dpa_rincian_mata_anggaran.jns_item = 'I'
```

@total = @anggaran + @rinci

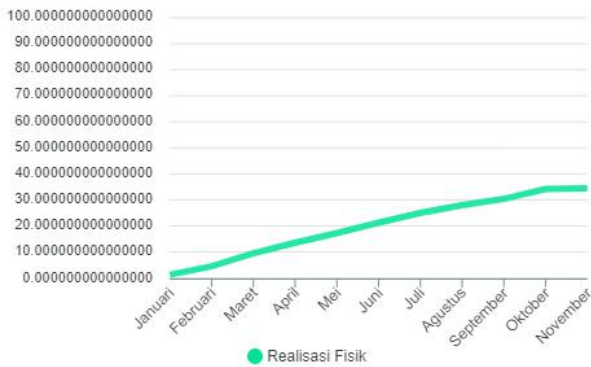
Setelah mendapatkan totalnya akan realisasi fisik akan dibagi dengan total dalam sebuah iterasi

```
SELECT @realisasi = SUM(realisasi_fisik)
FROM smep_realisasi
WHERE tahun = now() AND
bulan IN (1,2,3,4,5,6,7,8,9,10,11,12)
```

@hasil = @realisasi / @total



Grafik Realisasi Fisik



Gambar 7. Grafik Realisasi Fisik

D. Pengujian

Query yang sudah dibuat dan dijalankan tadi akan diuji apakah sudah berhasil apabila *query* yang dijalankan dapat berjalan dan sesuai dengan yang diharapkan. Selain itu perlu diukur juga berapa lama waktu eksekusi *query*.

Nama Fungsi	Berhasil / Gagal	Waktu Eksekusi
SKPD Serapan Tertinggi	Berhasil	1.56 Detik
SKPD Serapan Terendah	Berhasil	1.56 Detik
Total Anggaran	Berhasil	3.86 Detik
Total Serapan	Berhasil	3.04 Detik
Sisa Anggaran	Berhasil	4 Detik
Grafik Target & Realisasi Keuangan	Berhasil	1.31 Detik
Grafik Realisasi	Berhasil	2.43 Detik

Fisik		
-------	--	--

V. PEMBAHASAN

Berdasarkan apa yang sudah dijabarkan pada Bab Hasil, terdapat beberapa hal yang patut menjadi perhatian. Utamanya berkaitan dengan *query* yang membutuhkan perulangan. *Query* grafik dan rekapitulasi keuangan untuk mendapatkan data sampai dengan bulan tertentu perulangan masih dilakukan di *Laravel*. Agar hasilnya lebih efisien, maka bisa dibuat perulangan di dalam *query* dan disimpan dalam *stored procedure*.

VI. KESIMPULAN

Sistem Monitoring dan Evaluasi Pembangunan dapat dikembangkan dengan menggunakan Multi Join Query. Penerapan Multi Join Query umumnya sama dengan *join tabel*, yang membedakan hanya jumlah tabel dan tabel tersebut berjenjang.

REFERENCES

- [1] Bagian Administrasi Pembangunan Kota Probolinggo, "Daftar SKPD Kota Probolinggo," *pembangunan.probolinggokota.go.id* 2017. [Online]. Available: https://pembangunan.probolinggokota.go.id/web/kontent/63/data_skpd. [Accessed: 15-Nov-19]
- [2] N. A. Maghfiroh, "Efektivitas Sistem Informasi Pelaporan Monitoring Dan Evaluasi (E-Controlling) Pada Bagian Evaluasi Pembangunan Sekretariat Daerah Kota Bontang," *eJournal Unmul*, Vol 5, Nomor 1, pp. 36-37, 2016
- [3] O. K. Muslih and I. H. Saleh, "Structure Query Language Join Statement," *Journal of Computer Science*, Vol 6, Nomor 5, pp. 585, 2010
- [4] Dorota Wdzięczna, "An Illustrated Guide to Multiple Join", *academy.vertabelo.com/blog/illustrated-guide-multiple-join* 2017. [Online]. Available : <https://academy.vertabelo.com/blog/illustrated-guide-multiple-join> [Accessed: 3-Des-19]
- [5] Mohammad Alamery, Ahmad Faraahi, H. H. S. Javadi, Sadegh Nourossana, and Hossein Erfani, "Multi-Join Query Optimization Using the Bees Algorithm", *Advances in Intelligent and Soft Computing*, Vol 79, pp 449-457, 2010