

PERANCANGAN DATA WAREHOUSE PADA SOFTWARE LABORATORY CENTER

Alvin Chandra

Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Bina Nusantara
Jln. Kebon Jeruk Raya No. 27, Kebon Jeruk, Jakarta Barat 11530
alvin.chandra@binus.ac.id

ABSTRACT

The purpose of this study is to analyze the existing database on Software Laboratory Center to obtain necessary information and design data warehouse to integrate existing data to obtain some global information. The methods used are analysis and design method. The method of analysis is done by conducting surveys and analysis of the running system, analysis and identification the weaknesses of the running system, and troubleshooting analysis. And the data warehouse design method is done by applying the nine steps (Nine-Step Methodology) that used by Ralph Kimball's to design star schema. The result is data warehouse that provides some global information which global, relevant, and integrated which can be seen from various points of view that is useful for the leaders to make decisions. Separate data warehouse from operational databases that already exist required by Software Laboratory Center to assist leaders in making strategic decisions in a fast and precise way.

Keywords: Data warehouse, ETL, dimensi, fakta, database

ABSTRAK

Tujuan penelitian ini adalah menganalisis database yang ada pada Software Laboratory Center untuk mendapatkan informasi yang diperlukan dan merancang data warehouse untuk mengintegrasikan data-data yang ada sehingga didapatkan informasi yang bersifat global (menyeluruh). Adapun metode yang digunakan adalah metode analisis dan perancangan. Metode analisis dilakukan dengan mengadakan survey dan analisis atas sistem yang sedang berjalan, analisis kelemahan dan identifikasi permasalahan sistem yang berjalan, dan analisis pemecahan masalah. Sedangkan metode perancangan data warehouse dilakukan dengan menerapkan 9 langkah (Nine-Step Methodology) yang digunakan Ralph Kimball dalam merancang skema bintang. Hasil yang dicapai adalah data warehouse yang menyediakan informasi bersifat global, relevan, dan terintegrasi yang dapat dilihat dari berbagai macam sudut pandang sehingga berguna bagi para pimpinan untuk mengambil keputusan. Data warehouse yang terpisah dari database operational yang telah ada dibutuhkan oleh Software Laboratory Center untuk membantu para pimpinan dalam pengambilan keputusan-keputusan yang bersifat strategis secara cepat dan tepat.

Kata kunci: Data warehouse, ETL, dimensi, fakta, database

PENDAHULUAN

Seiring dengan perkembangan teknologi yang semakin pesat, *Software Laboratory Center* sebagai unit kerja dari Bina Nusantara University dituntut untuk mengikuti perkembangan tersebut. Dampaknya adalah persaingan bisnis global yang semakin pesat dan dinamis. Para eksekutif *Software Laboratory Center* dituntut untuk bijaksana dalam pengambilan kebijakan dan langkah-langkah yang menuntun Bina Nusantara University untuk mengembangkan bisnisnya. Untuk itu, dibutuhkan informasi-informasi yang terstruktur, lengkap, dan akurat. Penggabungan antara teknologi dengan informasi yang benar, akan menghasilkan keuntungan yang besar bagi *Software Laboratory Center*. Dengan adanya teknologi informasi, dapat dikembangkan suatu sistem yang dapat menyimpan data-data dalam suatu perusahaan yang dikenal dengan sistem basis data. Bagi perusahaan yang telah mengimplementasikan sistem basis data dalam proses bisnisnya, maka perusahaan tersebut akan mempunyai data yang besar juga. Data-data tersebut dapat menjadi dasar pertimbangan bagi para eksekutif dalam mengambil keputusan strategis. Namun, jumlah data yang besar tersebut menjadi kendala bagi para eksekutif untuk menganalisis informasi penting yang ada.

Untuk mengatasi masalah tersebut, maka diperlukan sebuah teknologi yang dapat mengelola informasi yang ada agar diperoleh berbagai keuntungan yang akan membawa kemajuan bagi *Software Laboratory Center*. Teknologi ini sering disebut dengan *data warehouse*. *Data warehouse* merupakan suatu basis data analisis yang dapat mendukung pengambilan keputusan. Dengan adanya *data warehouse*, berbagai data dalam suatu organisasi dapat diintegrasikan ke suatu tempat penyimpanan yang akan memberikan kemudahan bagi pemakai untuk membuat laporan dan melakukan analisis data. *Data warehouse* menyediakan pondasi bagi perusahaan yang memiliki sistem aplikasi yang belum terintegrasi. *Data warehouse* juga mengatur dan menyimpan data-data historis *Software Laboratory Center* yang diperlukan untuk menghasilkan informasi dan proses analisis dalam jangka panjang.

METODE

Untuk mendapatkan gambaran mengenai keadaan dan permasalahan yang sedang dihadapi oleh perusahaan, diperlukan data-data yang tepat untuk keperluan analisis dan perancangan aplikasi *data warehouse* ini. Oleh karena itu, dilakukan berbagai penelitian pada Bina Nusantara University bagian *Software Laboratory Center* melalui tahap:

- a. *Survey* atas sistem yang berjalan.
- b. Analisis sistem yang berjalan.
- c. Analisis kelemahan sistem berjalan.
- d. Identifikasi permasalahan sistem berjalan.
- e. Analisis pemecahan masalah

Pengenalan Data warehouse

Data warehouse adalah sekumpulan data yang diperoleh dari berbagai sumber yang digunakan untuk mendukung proses pembuatan keputusan manajemen di dalam perusahaan. Beberapa karakteristik *data warehouse* menurut Inmon (2002: 31) antara lain:

a. *Subject Oriented*

Data warehouse disusun berdasarkan subjek-subjek utama suatu *database* (seperti pelanggan, produk, penjualan) dimana setiap area fisik subjek diimplementasikan sebagai kumpulan dari *table* yang berhubungan dalam *data warehouse* dan tidak berorientasi pada proses atau fungsi aplikasi

tertentu. Orientasi subjek ini berbeda dengan *Online Transaction Processing (OLTP)*. Berikut tabel perbandingan sistem OLTP dengan *Data Warehouse* (Connolly & Begg, 2002: 1049).

Tabel 1 Tabel Perbandingan Sistem OLTP dan Sistem *Data Warehouse*

Sistem OLTP	Sistem <i>data warehouse</i>
Menangani data sekarang	Menangani data historis
Menyimpan <i>detailed data</i>	Menyimpan <i>detailed, lightly, dan highly summarized data</i>
Data bersifat dinamis	Data bersifat statis
Proses berulang	Proses sewaktu-waktu tidak terstruktur dan <i>heuristic</i>
Jumlah transaksi tinggi	Jumlah transaksi rendah sampai sedang
<i>Transaction driven</i>	<i>Analysis driven</i>
Berorientasi aplikasi	Berorientasi subjek
Mendukung keputusan harian	Mendukung keputusan strategis
Melayani banyak <i>user</i>	Melayani sedikit <i>user</i> (manajerial)

b. Integrated

Dari semua aspek *data warehouse*, integrasi adalah yang paling penting. Data diambil dari banyak sumber ke suatu *data warehouse*. Data tersebut diubah, diformat ulang, disusun ulang, diringkas, dan seterusnya. Hasilnya, ketika data tersebut sampai di *data warehouse*, data tersebut mempunyai satu gambaran fisik tunggal.

c. Time Variant

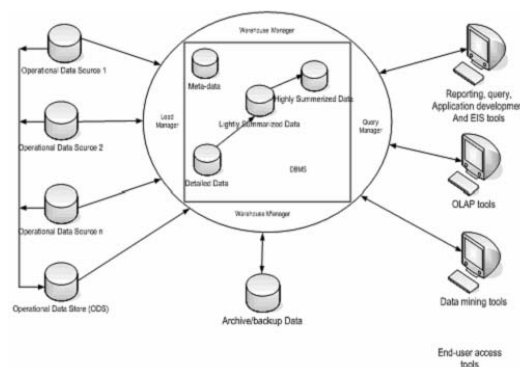
Data warehouse menggunakan *time stamp* untuk merepresentasikan data historis. Dimensi waktu sangat kritis untuk mengidentifikasi *trend*, memprediksi operasi-operasi mendatang, dan mengatur sasaran-sasaran yang beroperasi.

d. Non Volatile

Tidak seperti halnya *record* pada *database* operasional yang biasanya selalu diakses dan dimanipulasi, data pada *data warehouse* mempunyai karakteristik yang berbeda. Data pada *data warehouse* di-load dan diakses oleh *user* yang bersangkutan, tetapi tidak dapat dan tidak boleh di-update. Sebagai gantinya, ketika data pada *data warehouse* di-load, data tersebut di-load dalam format statis. Nantinya, ketika suatu perubahan terjadi, *record* statis baru akan disimpan. Dengan begitu, *record* lampau dari data tersebut tetap tersimpan di *data warehouse*.

Arsitektur *data warehouse* merupakan suatu kerangka yang dirancang dengan cara memahami bagaimana data dipindahkan di dalam sistem. Karakteristik arsitektur dari *data warehouse* adalah:

- Data diambil dari sistem informasi yang telah ada, *database*, dan *file*.
- Data tersebut diintegrasikan dan ditransformasikan sebelum disimpan ke dalam *data warehouse*.
- Data warehouse* adalah *read-only database* yang diciptakan untuk mengambil keputusan.
- User* mengakses *data warehouse* melalui *front-end tool* atau aplikasi.



Gambar 1 Gambar Arsitektur *Data warehouse* (Connolly & Begg, 2002: 1053)

Komponen utama *data warehouse* menurut Connolly dan Begg (2002: 1053) adalah:

- a. *Operational Data Source*, sumber data ini didapat dari data operasional yang dilakukan pada *database* awal.
- b. *Operational Data Store (ODS)*, tempat penyimpanan data operasional yang sedang terjadi dan yang terintegrasi dimana digunakan untuk analisa.
- c. *Load Manager* (sering juga disebut komponen *front-end*), menampilkan semua operasi yang diasosiasikan dengan data yang telah diekstrak dan di-load ke dalam *warehouse*.
- d. *Warehouse Manager*, menampilkan semua operasi yang diasosiasikan dengan manajemen data dalam *warehouse*.
- e. *Query Manager* (disebut juga dengan komponen *back-end*), menampilkan semua operasi yang diasosiasikan dengan manajemen dari *user queries*.
- f. *Detailed Data, Meta-data, Lightly and Highly Summarized Data*, untuk komponen ini sudah dijelaskan pada bagian struktur *data warehouse*.
- g. *Archive/ Backup Data*, area *warehouse* yang menyimpan *detailed* dan *summarized data* yang bertujuan sebagai arsip dan *backup data*.
- h. *End-User Access Tools*, dapat dikategorikan menjadi lima grup utama: *data reporting and query tools, application development tools, executive information systems (EIS) tools, online analytical processing (OLAP) tools and data mining tools*.

Fungsi Data warehouse

Menurut Williams (1998: 533) *data warehouse* mempunyai kegunaan sebagai berikut:

a. Pembuatan Laporan

Pembuatan laporan merupakan salah satu kegunaan *data warehouse* yang paling umum. Dengan menggunakan *query* sederhana dalam *data warehouse*, dapat dihasilkan informasi per tahun, per kuartal, per bulan dan bahkan per hari.

b. OLAP

Data warehouse digunakan dalam melakukan analisis bisnis untuk mengetahui kecenderungan pasar dan faktor-faktor penyebabnya, karena dengan adanya *data warehouse*, semua informasi baik detil maupun hasil *summary* yang dibutuhkan dalam proses analisis mudah didapat. Dalam hal ini *data warehouse* merupakan *tool* yang handal untuk analisa data yang kompleks.

c. Data Mining

Data Mining adalah proses untuk mencari informasi dan pengetahuan baru dengan cara menggali (*mining*) data yang berjumlah banyak pada *data warehouse* dengan menggunakan kecerdasan buatan (*artificial intelligence*), statistik, dan matematika. *Data mining* merupakan teknologi yang diharapkan bisa menjembatani komunikasi antar data dan pemakainya.

d. Proses Informasi Eksekutif

Data warehouse digunakan untuk mencapai ringkasan informasi yang penting dengan tujuan membuat keputusan bisnis, tanpa harus menjelajahi keseluruhan data. Dengan menggunakan *data warehouse*, segala laporan telah diringkaskan dan dapat pula diketahui rinciannya secara lengkap, sehingga mempermudah proses pengambilan keputusan. Informasi dan data pada laporan *data warehouse* menjadi target informatif bagi *user*, yang dalam hal ini adalah pihak eksekutif.

Keuntungan Data warehouse

Berdasarkan Connolly & Begg (2002: 1048), pengimplementasian yang sukses dari sebuah *data warehouse* dapat memberikan keuntungan besar bagi perusahaan, yaitu:

- a. Keuntungan potensial yang besar dalam investasi.
- b. Keuntungan yang kompetitif.
- c. Meningkatnya produktivitas dari pengambil keputusan perusahaan

Teori Permodelan Data warehouse

a. Skema Bintang

Menurut Creative Data, Skema Bintang adalah suatu teknik standar dalam merancang tabel ringkasan dari *data warehouse*. Skema bintang terdiri dari sebuah tabel data sentral, atau tabel fakta yang terhubung dengan satu atau lebih tabel dimensi. Disebut skema bintang karena model ini menyerupai bintang, dengan titik-titik yang berpusat dari tengah. Pusat dari skema bintang terdiri dari satu atau lebih tabel fakta dan titik-titik pada skema merupakan tabel dimensi yang memuat informasi pada atribut tertentu dalam tabel fakta.



Gambar 2 Skema Bintang

Skema bintang menjadi standar perancangan *data warehouse* karena keuntungan-keuntungan yang dimilikinya, yaitu sebagai berikut:

- Menciptakan desain *database* yang memberikan waktu respon yang cepat.
- Menyediakan pendesainan yang dapat dengan mudah dimodifikasi atau ditambahkan sesuai dengan perkembangan dan pertumbuhan *data warehouse*.
- Meningkatkan kinerja dimana setiap tabel dimensi dapat diindeks tanpa menimbulkan perancangan ruang pada level *database*.
- Selain mengurangi jumlah penggabungan secara fisik (*join physical*) juga mengizinkan pengguna untuk mendefinisikan hirarki dan mengarahkan antar tabel.
- Pendesainan data secara paralel sehingga pengguna dapat memandang dan menggunakan data secara bersamaan.
- Memudahkan pemahaman dan pengaturan *metadata* dari segi pengembang dan pengguna.
- Memperluas pilihan dari alat pengaksesan *data front-end* seperti beberapa produk yang perlu pendesainan skema bintang.

b. Skema Snowflake

Snowflake merupakan variasi lain dari skema bintang dimana tabel dimensi dari skema bintang diorganisasikan menjadi suatu hirarki dengan melakukan normalisasi. Prinsip dasar dari skema ini tidak jauh berbeda dari skema bintang. Penggunaan tabel dimensi sangatlah mendasar, karena

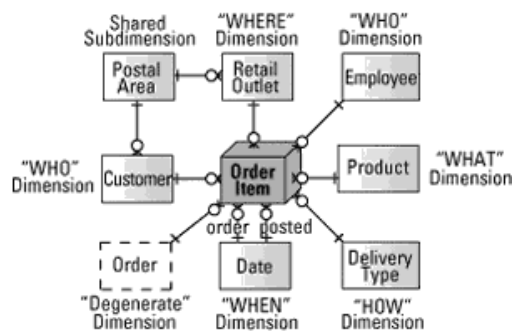
itulah perbedaan mendasar dari skema bintang dan skema *snowflake*. Skema *snowflake* menggunakan beberapa tabel fakta dan tabel dimensi yang sudah mengalami normalisasi, sedangkan skema bintang menggunakan tabel dimensi yang masih denormalisasi. Skema *snowflake* dibuat berdasarkan OLTP sehingga semua data akan termuat *detail* dalam setiap tabel fakta dan tabel dimensi. Keuntungan dari skema *snowflake*:

- Kecepatan memindahkan data dari data OLTP ke dalam *Metadata*.
- Sebagai kebutuhan dari alat pengambil keputusan tingkat tinggi dimana dengan tipe yang seperti ini seluruh struktur dapat digunakan sepenuhnya.
- Banyak yang beranggapan lebih nyaman merancang dalam bentuk normal ketiga.

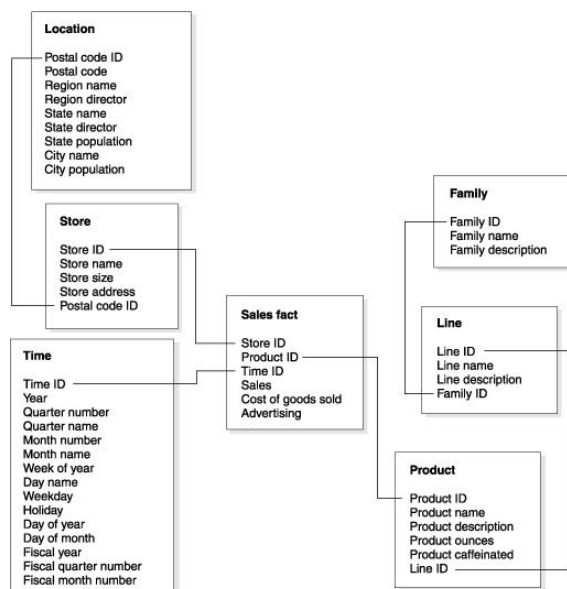
Sedangkan kerugiannya adalah mempunyai masalah yang besar dalam hal kinerja, hal ini disebabkan karena semakin banyaknya *join* antar tabel–tabel yang digunakan dalam *snowflake*, maka kinerja juga semakin lambat lihat Gambar 3.

c. Skema Starflake

Menurut Connolly dan Begg (2002: 1081), skema *starflake* merupakan struktur *hybrid* yang berisi kombinasi antara skema bintang yang telah didenormalisasi dan *snowflake* yang telah dinormalisasi. Beberapa dimensi dapat menggunakan bentuk tertentu untuk memenuhi berbagai kebutuhan *query* lihat Gambar 4.



Gambar 3 Skema Snowflake



Gambar 4 Skema Starflake

Perancangan Data warehouse

Untuk mendapatkan gambaran mengenai keadaan dan permasalahan yang sedang dihadapi oleh perusahaan, diperlukan data-data yang tepat untuk keperluan analisis dan perancangan aplikasi *data warehouse* ini. Oleh karena itu, dilakukan berbagai penelitian pada Bina Nusantara University bagian *Software Laboratory Center* melalui tahap:

1. Identifikasi Proses

Pada tahap ini dilakukan penentuan subjek dari permasalahan yang sedang dihadapi. Berdasarkan hasil analisis ditemukan beberapa proses penting dalam kegiatan operasional *Software Laboratory Center*, yaitu:

- Perekrutan karyawan baru
- Pemantauan kinerja karyawan *Software Laboratory Center*.
- Transaksi praktikum.

2. Identifikasi Grain

Memilih *grain* berarti memutuskan secara tepat apa yang digambarkan oleh *record* dalam tabel fakta. Setelah menentukan *grain* tabel fakta, dimensi untuk setiap tabel fakta dapat diidentifikasi. *Grain* dalam perancangan *data warehouse* ini meliputi:

- Laporan Praktikum yang terdiri dari : Laporan Nilai Mahasiswa, Laporan Absensi, Laporan Kelulusan, Laporan Utilisasi Ruang Praktikum
- Laporan Karyawan yang terdiri dari : Laporan IKAA, Laporan Koreksi, Laporan Pembuatan Soal, Laporan Perubahan Nilai.
- Laporan Trainee yang meliputi : Laporan Kehadiran Trainee, Laporan Nilai Trainee.

3. Identifikasi Dimensi

Dalam tahapan ini yang dilakukan adalah identifikasi dimensi dengan detail yang secukupnya untuk mendeskripsikan sesuatu. Hubungan ini biasanya dibuat dalam bentuk table-tabel. Berikut adalah salah satu contohnya:

Tabel 2 Tabel *Grain* VS Dimensi Pada Laporan Absensi

<i>Grain</i>	Persentase Kehadiran	Persentase Absen
Dimensi		
Waktu	X	X
Angkatan	X	X
Mata Kuliah	X	X
Kelas	X	X

4. Identifikasi Fakta

Memilih fakta yang akan digunakan dalam *data mart*. Masing-masing fakta memiliki data yang dapat dihitung yang kemudian akan ditampilkan dalam bentuk laporan, grafik atau diagram lainnya. Berikut ini adalah fakta-fakta yang terdapat dalam *data warehouse* yaitu:

- Fakta Laporan Nilai Mahasiswa
- Fakta Laporan Absensi
- Fakta Laporan Kelulusan
- Fakta Laporan Utilisasi Ruang Praktikum
- Fakta Laporan IKAA
- Fakta Laporan Koreksi
- Fakta Laporan Pembuatan Soal
- Fakta Laporan Perubahan Nilai
- Fakta Laporan Kehadiran Trainee
- Fakta Laporan Nilai Trainee

5. Menyimpan *Pre-Calculation* Dalam Tabel Fakta

Pada saat fakta-fakta telah dipilih, setiap fakta harus diperiksa ulang untuk menentukan apakah terdapat kesempatan dalam menggunakan *pre-calculation*. Berikut ini salah satu contoh kalkulasi awal yang disimpan dalam tabel fakta, yaitu Fakta_LaporanKoreksi, sebagai berikut:

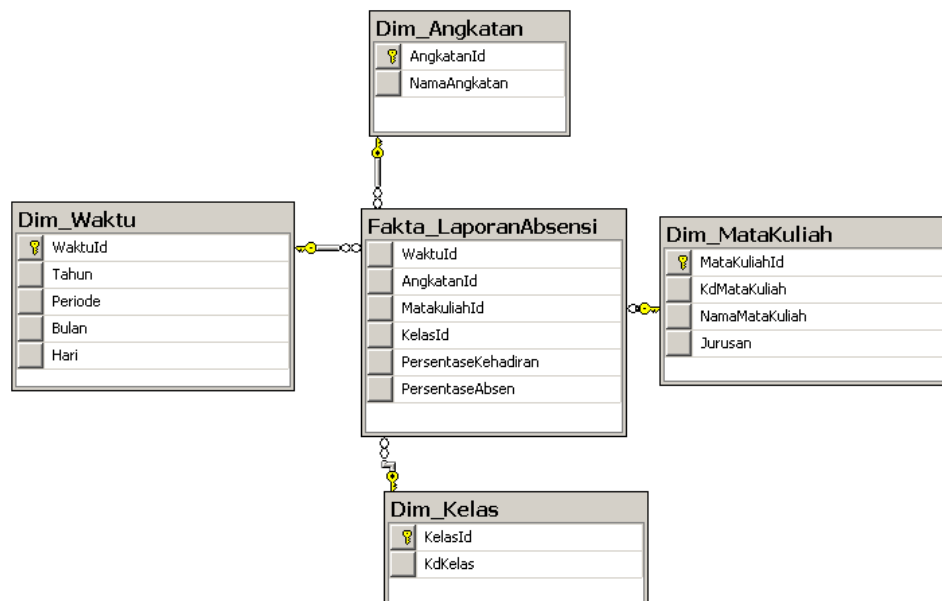
- TM yang merupakan jumlah hari telat pengoreksian jawaban TM dari setiap karyawan.
- Proyek yang merupakan jumlah hari telat pengoreksian hasil Proyek dari setiap karyawan.
- UTP yang merupakan jumlah hari telat pengoreksian jawaban UTP dari setiap karyawan.
- UAP yang merupakan jumlah hari telat pengoreksian jawaban UAP dari setiap karyawan.
- JumlahHariTelat yang merupakan total hari telat seluruh pembuatan soal dari setiap karyawan.
- PersentaseKetepatanWaktu yang merupakan persentase ketepatan waktu karyawan dalam pembuatan soal.

6. *Rounding Out* Tabel Dimensi

Pada tahap ini kembali ke tabel dimensi dan menambahkan deskripsi teks sebanyak mungkin ke dimensi. Deskripsi teks harus dapat dimengerti oleh *user*. Dalam tahap ini mencakup informasi berupa Daftar Deskripsi Tabel Dimensi beserta Daftar Tabelnya, dan Gambar skema yang sudah dirancang. Dalam perancangan *data warehouse Software Laboratory Center* ini dihasilkan sepuluh skema bintang sebagai berikut:

- a. Skema bintang Fakta_LaporanNilaiMahasiswa
- b. Skema bintang Fakta_LaporanAbsensi
- c. Skema bintang Fakta_LaporanKelulusan
- d. Skema bintang Fakta_LaporanUtilisasiRuangPraktikum
- e. Skema bintang Fakta_LaporanIKAA
- f. Skema bintang Fakta_LaporanKoreksi
- g. Skema bintang Fakta_LaporanPembuatanSoal
- h. Skema bintang Fakta_LaporanPerubahanNilai
- i. Skema bintang Fakta_LaporanKehadiranTrainee
- j. Skema bintang Fakta_LaporanNilaiTrainee

Berikut adalah salah satu contoh gambar skema bintang untuk Fakta_LaporanAbsensi:



Gambar 5 Gambar Skema Bintang Fakta_LaporanAbsensi

7. Memilih Durasi Database

Menentukan batas waktu dari umur data yang diambil dan akan dipindahkan ke dalam tabel fakta. Misalnya, data *Software Laboratory Center* dua tahun lalu atau lebih diambil dan dimasukkan dalam tabel fakta.

Tabel 3 Tabel Durasi Basis Data

Nama Aplikasi	Database	Database ada sejak tahun	Data yang masuk ke data warehouse	Data Dalam data warehouse
<i>Software</i>	SLC	2001	September 2004 – Agustus 2008	3 tahun 11 bulan

8. Mengawasi Perubahan Dimensi

Adapun perubahan dimensi dapat ditanggulangi dengan tiga macam cara, yaitu mengganti secara langsung pada tabel dimensi, pembentukan *record* baru untuk setiap perubahan baru, dan perubahan data yang membentuk kolom baru yang berbeda. Dalam perancangan ini dipilih cara kedua yaitu jika ada perubahan data, maka data baru akan dimasukkan sebagai *record* baru dan *record* lama tetap ada dan tidak dihapus. Dalam analisis yang dilakukan, kemungkinan terjadinya perubahan data adalah pada nama-nama yang ada di dalam tabel dimensi. Sebagai contoh, alamat karyawan jika sewaktu-waktu alamat karyawan berubah. Nama karyawan, jika ternyata ada kesalahan dalam memasukkan nama karyawan, dan lain sebagainya.

9. Memutuskan Prioritas Mode Query

Dalam tahap ini dilakukan penentuan waktu proses ETL (*Extract, Transform, and Load*), *backup* yang dilakukan secara berkala, dan keamanan penggunaan laporan serta analisis kapasitas media penyimpanan dan pertumbuhan data, termasuk keterangan pelaku dan jangka waktu atau jadwal proses. Untuk ETL sendiri adalah proses-proses dalam *data warehouse* yang meliputi:

- Mengekstrak data dari sumber-sumber eksternal.
- Mentransformasikan data ke bentuk yang sesuai dengan keperluan.
- Memasukkan data ke target akhir, yaitu *data warehouse*.

ETL merupakan proses yang sangat penting, dengan ETL data dapat dimasukkan ke dalam *data warehouse*. ETL juga dapat digunakan untuk mengintegrasikan data dengan sistem yang sudah ada sebelumnya. Tujuan ETL adalah mengumpulkan, menyaring, mengolah, dan menggabungkan data-data yang relevan dari berbagai sumber untuk disimpan ke dalam *data warehouse*. Hasil dari proses ETL adalah dihasilkannya data yang memenuhi kriteria *data warehouse* seperti data yang historis, terpadu, terangkum, statis, dan memiliki struktur yang dirancang untuk keperluan proses analisis. Untuk proses backup dan penggunaan laporan dapat dilihat pada table berikut:

Tabel 4 Tabel Proses Backup

Pelaku Backup	Dilakukan setiap	Keterangan
Karyawan DBA	Akhir semester sebelum proses ETL selanjutnya dilakukan	<i>Backup data warehouse</i> dilakukan untuk menanggulangi proses ETL yang gagal.

Tabel 5 Tabel Penggunaan Laporan Atau Kebutuhan Informasi

User	Informasi
SLC Manager	Informasi digunakan untuk menentukan segala sesuatu yang berhubungan dengan kegiatan praktikum di <i>Software Laboratory Center</i> dan untuk mengambil keputusan-keputusan yang strategis.
Operational Support Section	Informasi digunakan untuk mengetahui utilisasi sarana praktikum agar dapat

Head	dilakukan maintenance dan pengembangan sarana praktikum untuk menunjang proses kegiatan praktikum yang lebih baik.
Research Development and Training Section Head	Informasi digunakan untuk meningkatkan perkembangan dari karyawan dan matakuliah praktikum baik melalui peningkatan pelatihan ataupun fasilitas yang dapat menunjang proses kegiatan pelatihan karyawan dan kegiatan praktikum.
Academic and Operation Section Head	Informasi digunakan untuk mengevaluasi indeks kinerja karyawan dan seluruh kegiatan operasional praktikum agar dapat ditentukan dan dilaksanakan jenis pengembangan pelatihan karyawan dan SAP praktikum bersama bagian <i>Research Development and Training</i> .

Kemudian, di dalam proses OLTP, kapasitas media penyimpanan menjadi salah satu faktor yang sangat penting. Proses *insert*, *update*, dan *delete* akan mempengaruhi pertumbuhan data secara terus menerus. Demikian pula pada perancangan *data warehouse*, harus dilakukan analisis pertumbuhan data untuk memperoleh perkiraan kapasitas media penyimpanan yang memadai dalam menampung data hingga beberapa tahun ke depan. Rumus yang akan digunakan untuk perhitungan kebutuhan penyimpanan *record* dalam SQL Server 2005 (SQL Server Books Online) adalah:

- Num_Rows = Jumlah Baris / Jumlah *Record*
- Num_Cols = Jumlah Kolom
- Fixed_Data_Size = Jumlah byte yang dibutuhkan seluruh kolom *fixed-length*
- Num_Variable_Cols = jumlah kolom *variable-length*
- Max_Var_Size = Jumlah byte maksimum dari seluruh kolom *variable-length*
- Null_Bitmap = $2 + ((\text{Num_Cols} + 7) / 8)$
- Variable_Data_Size = $2 + (\text{Num_Variable_Cols} \times 2) + \text{Max_Var_Size}$
- Row_Size = Fixed_Data_Size + Variable_Data_Size + Null_Bitmap + 4
- Rows_Per_Page = $8096 / (\text{Row_Size} + 2)$
- Num_Pages = Num_Rows / Rows_Per_Page
- Heap size (bytes) = $8192 \times \text{Num_Pages}$

Analisis perkiraan kapasitas media penyimpanan data pada *data warehouse Software Laboratory Center* adalah seperti berikut ini, dimana 'n' adalah variabel tahun. Perkiraan ini dibatasi sampai 5 tahun mendatang. Berikut adalah perhitungan pertumbuhan data untuk seluruh tabel fakta dengan asumsi tingkat pertumbuhan masing-masing adalah 10%.

$$R_n = R \times (n + (1 + i)^n)$$

R = jumlah *record*

n = tahun

I = persentase pertumbuhan *record*

Tabel 6 Tabel Analisis Kapasitas Media Penyimpanan Untuk Tabel Fakta

Nama Tabel	Besar 1 <i>Record</i> (Bytes)	Jumlah <i>Current</i> <i>Record</i>	Jumlah <i>Record</i> Sampai 5 Tahun ke Depan	Jumlah Bytes (Mbytes)
FaktaLaporanNilaiMahasiswa	121.125	7560	249877	29.6
FaktaLaporanAbsensi	41.625	83160	2748650	115.5
FaktaLaporanKelulusan	29.5	7560	249877	7.6
FaktaLaporanUtilisasiRuangPraktikum	31.175	181440	5997055	192.8
FaktaLaporanIKAA	131	3360	111057	14.3
FaktaLaporanKoreksi	60.375	22680	749632	45.05
FaktaLaporanPembuatanSoal	60.375	6720	222114	13.35
FaktaLaporanPerubahanNilai	33.625	378	12494	0.43
FaktaLaporanKehadiranTrainee	29.5	371	12263	0.375
FaktaLaporanNilaiTrainee	37.625	371	12263	0.46
Total	575.925	313600	10365282	419.475

Tabel 7 Tabel Analisis Kapasitas Media Penyimpanan Untuk Tabel Dimensi

Nama Tabel	Besar Record (Bytes)	Jumlah Current Record	Jumlah Record Sampai 5 Tahun ke Depan	Jumlah Bytes (Mbytes)
Dim_Ruang	20.25	45	46	0
Dim_Karyawan	21.375	280	580	0.02
Dim_Angkatan	17.125	12	60	0
Dim_Group	14.125	4	4	0
Dim_AstSpv	18.125	24	44	0
Dim_SubCo	18.125	16	36	0
Dim_MataKuliah	80.25	70	75	0
Dim_Kelas	18.125	294	394	0
Dim_Shift	17.125	7	7	0
Dim_Waktu	29.875	117	1942	0.06
Dim_Trainee	81.875	400	930	0.08
Total	336.375	1269	4118	0.16

Berdasarkan hasil analisis kapasitas media penyimpanan data diatas dapat diperkirakan kebutuhan penyimpanan untuk *data warehouse Software Laboratory Center* adalah 419.635 Mbytes untuk jangka waktu 5 tahun.

HASIL DAN PEMBAHASAN

Evaluasi Perangkat

Perangkat keras yang disarankan untuk digunakan dalam implementasi *data warehouse* agar memperoleh performa optimal adalah:

- *Server* (Prosesor Intel Xeon 2.8 GHz, *Harddisk* 140 GB, *Memory* 1 GB, DVD-RW).
- *Client* (Prosesor Intel Core2Duo 1.8 GHz, *Harddisk* 80 GB, *Memory* 1 GB, DVD-RW).

Sedangkan untuk perangkat lunak yang disarankan dalam menjalankan aplikasi *data warehouse* pada *Software Laboratory Center* adalah:

- *Server*
 - Sistem operasi: Microsoft Windows XP Professional Editon Sp. 2
 - *Database*: Microsoft SQL Server 2005
- *Client*
 - Sistem operasi: Microsoft Windows XP Professional Edition Sp. 2
 - *Database*: Microsoft SQL Server 2005
 - *front end Tool*: Microsoft Excel 2007

Estimasi Waktu

Untuk melakukan implementasi sistem *data warehouse* akan dilakukan berbagai kegiatan untuk menunjang terlaksananya implementasi sistem dengan baik. Tabel dibawah adalah jadwal kegiatan yang dilakukan dalam perencanaan implementasi sistem *data warehouse* yang telah dirancang.

Tabel 8 Tabel estimasi waktu implementasi *data warehouse*

No	Kegiatan	Hari							
		1	2	3	4	5	6	7	8
1.	Instalasi aplikasi								

2. Transformasi data
3. Uji coba aplikasi
4. Pelatihan *user*
5. Evaluasi

Berikut penjelasan kegiatan-kegiatan pengimplementasian *data warehouse* diatas:

1. Instalasi aplikasi

Kegiatan ini adalah instalasi seluruh *software* yang dibutuhkan untuk mengimplementasi sistem *Data warehouse*, yaitu aplikasi *software* Microsoft SQL Server 2005 pada *server* dan *client* serta Microsoft Excel 2007 pada *client*. Hal ini perlu dilakukan karena belum semua *software* yang dibutuhkan untuk implementasi sudah terinstalasi.

2. Transformasi data

Kegiatan ini merupakan proses *restore data warehouse* ke dalam DBMS *Software Laboratory Center* karena DBMS yang digunakan untuk perancangan *data warehouse* berbeda dengan DBMS *Software Laboratory Center*. Karena itu dilakukan *backup* dari DBMS yang digunakan untuk perancangan *data warehouse* terlebih dahulu, kemudian di *restore* ke dalam DBMS *Software Laboratory Center*. Kemudian dilakukan proses *extract, transform, dan load* seluruh data dari *database* operasional ke dalam *data warehouse*.

3. Uji coba aplikasi

Setelah semua syarat dan kebutuhan perangkat keras dan perangkat lunak maka dilakukan uji coba terhadap aplikasi *Data warehouse* yang telah dibuat. Dalam uji coba ini akan dianalisis apakah aplikasi yang dibuat sesuai dengan kebutuhan, bebas bug, ataupun kesalahan yang dapat mengganggu performa sistem.

4. Pelatihan *user*

Kegiatan ini dilakukan agar *user* yang menggunakan aplikasi nantinya dapat beradaptasi dan mengenal aplikasi baru tersebut dengan baik sehingga pengoperasian aplikasi dapat berjalan dengan baik.

5. Evaluasi

Kegiatan ini adalah evaluasi terhadap sistem yang diimplementasikan maupun evaluasi terhadap kegiatan pelatihan *user* yang telah dilakukan. Kegiatan ini dilakukan untuk mengukur tingkat keberhasilan sistem maupun *user* dengan menggunakan berbagai indikator.

Rencana Backup

Untuk melakukan implementasi sistem *data warehouse* maka akan dilakukan kegiatan *backup* untuk menunjang terlaksananya implementasi sistem dengan baik. Berikut jadwal kegiatan *backup* sistem *data warehouse*:

Tabel 9 Tabel rencana *backup*

Proses Backup	Jadwal
OLTP <i>data warehouse</i>	<i>Backup</i> OLTP dilakukan setiap hari minggu. <i>Backup data warehouse</i> dilakukan setiap akhir semester sebelum proses ETL selanjutnya dilakukan.
<i>data Transformation Package</i>	<i>Backup</i> terhadap <i>package</i> ini dapat dilakukan pada waktu implementasi sistem untuk pertama kali.
Aplikasi	<i>Backup</i> aplikasi dapat dilakukan pada waktu implementasi sistem untuk pertama kali.

SIMPULAN

Dalam penulisan skripsi ini telah dilakukan analisis terhadap sistem yang berjalan pada *Software Laboratory Center*. Berdasarkan hasil analisis ditemukan beberapa permasalahan pada sistem *Software Laboratory Center*, yaitu belum adanya *data warehouse* sehingga menyulitkan *manager* dalam memperoleh informasi yang bersifat menyeluruh dan mengambil keputusan. Permasalahan itulah yang menjadi alasan dibuatnya *data warehouse*. Adapun dari penulisan ini dapat ditarik beberapa simpulan sebagai berikut.

1. Dengan dibuatnya *data warehouse* pada *Software Laboratory Center* pengaksesan informasi menjadi lebih mudah karena data yang dibutuhkan sudah tersedia dan dapat dilihat dari segala sudut pandang.
2. *Data warehouse* menghasilkan data yang konsisten, valid, dan akurat dalam bentuk laporan sehingga *manager* tidak perlu membaca dan menganalisis data dalam waktu yang cukup lama dan tentunya hal ini sangat mempermudah *manager* dalam mengambil keputusan secara cepat.

Adapun saran-saran yang harus diperhatikan agar implementasi *data warehouse* pada *Software Laboratory Center* berjalan dengan lancar adalah sebagai berikut:

1. Perlu dilakukan pemeliharaan (*maintenance*) *data warehouse* secara teratur dalam jangka waktu tertentu agar *data warehouse* dapat berjalan secara optimal.
2. Adanya pengawasan ketat pada saat transformasi data untuk memastikan data yang masuk ke dalam *data warehouse* adalah data yang konsisten, valid, dan akurat.

DAFTAR PUSTAKA

- Connolly, T., & Begg, C. (2002). *Database System: A Practical Approach to Design, Implementation, and Management*. Edisi ke-3. USA: Addison-Wesley.
- Inmon, W.H. (2002). *Building the Data Warehouse*, Third Edition. New York: John Wiley & Sons, Inc..
- Williams, C. (1998). *Professional Visual Basic 6 Databases*. Chicago: Wrox Press Ltd.