

PEMODELAN DATA WAREHOUSE PADA JURUSAN TEKNIK INFORMATIKA UNIKOM

DIAN DHARMAYANTI, ADAM MUKHARIL BACHTIAR, ANDRI HERYANDI
Program Studi Teknik Informatika – Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia

Informasi sangat dibutuhkan tidak hanya sebagai hasil rekapitulasi saja akan tetapi suatu informasi dapat digunakan untuk membantu dalam proses pengambilan keputusan bagi pihak manajerial maupun eksekutif. Program Studi Teknik Informatika ketika akan melaksanakan akreditasi menghadapi kesulitan dalam menghimpun informasi dikarenakan penyajian informasi didapat dari berbagai basis data dan file eksternal. Basis data dan file eksternal yang digunakan belum mempunyai struktur yang sama sehingga diperlukan lagi usaha untuk menyeragamkan data. Data warehouse adalah sebuah koleksi data yang berorientasi subjek, diintegrasikan, time-variant, dan non volatile untuk mendukung proses pembuatan manajemen pengambilan keputusan. Hasil dari data warehouse merupakan informasi hasil intisari dari berbagai macam basis data. Hasil penelitian menghasilkan fakta bahwa atribut data pada diagram relasi OLTP masih belum bisa memenuhi kebutuhan data yang ada pada diagram relasi *data warehouse* dengan membandingkan antara diagram relasi OLTP dengan diagram relasi *data warehouse*. Terdapat kekurangan data pada OLTP Program Studi Teknik Informatika UNIKOM yang mengakibatkan data pada *data warehouse* tidak bisa diisi. Dari hasil penelitian ini diharapkan nantinya akan ada integrasi data pada seluruh basis data yang berhubungan dengan *data warehouse* agar model *data warehouse* yang telah dibentuk dapat diimplementasi pada penelitian berikutnya

Kata Kunci - Basis Data, File eksternal, OLTP, *Data warehouse*

PENDAHULUAN

Perkembangan teknologi informasi telah banyak mempengaruhi perilaku suatu organisasi atau institusi dalam mengolah informasi. Penggunaan aplikasi untuk mengolah informasi merupakan salah satu bentuk perubahan perilaku di suatu organisasi atau institusi. Sebagai contoh, banyak perusahaan yang membuat sistem informasi untuk mengolah data yang mereka miliki dalam rangka mempercepat

pembuatan rekapitulasi baik dalam bentuk laporan maupun dalam bentuk grafik. Seiring dengan berkembangnya kebutuhan informasi, penggunaan aplikasi bukan lagi menjadi “obat” yang tepat untuk mengolah informasi. Informasi yang dibutuhkan tidak hanya berupa rekapitulasi saja akan tetapi dibutuhkan suatu informasi yang bisa digunakan untuk membantu mengambil keputusan bagi pihak manajerial maupun eksekutif.

Salah satu contohnya, saat jurusan Teknik

Informatika yang akan melaksanakan akreditasi menghadapi kesulitan dalam menghimpun informasi. Hal ini disebabkan karena penyajian informasi didapat dari berbagai basis data dan file eksternal. Selain itu basis data dan file eksternal yang digunakan belum mempunyai struktur yang sama sehingga diperlukan lagi usaha untuk menyeragamkan data.

Data warehouse adalah sebuah koleksi data yang berorientasi subjek, diintegrasikan, time-variant, dan non volatile untuk mendukung proses pembuatan manajemen pengambilan keputusan [1]. Hasil dari data warehouse merupakan informasi hasil intisari dari berbagai macam basis data. Berdasarkan kondisi di atas maka sebuah data warehouse bisa dijadikan solusi untuk memecahkan masalah di atas.

TINJAUAN PUSTAKA

Pengertian Data Warehouse

Data warehouse adalah sebuah tempat penyimpanan data yang lengkap dan konsisten dimana data yang disimpan itu saling berelasi dan dirancang berdasarkan query dan analisis daripada proses transaksinya[2]. Data tersebut kemudian ditransformasikan menjadi sebuah informasi yang dapat diakses kapan saja dan selalu up-to-date. Informasi ini kemudian akan digunakan untuk dianalisis untuk menghasilkan keputusan yang tepat.

Data warehouse juga bisa dikatakan sebagai kumpulan data yang berorientasi subjek, terintegrasi, tidak dapat di update, memiliki dimensi waktu yang digunakan untuk mendukung proses manajemen pengambilan keputusan dan kecerdasan bisnis. Berdasarkan definisi tersebut, maka data warehouse memiliki karakteristik [1] sebagai berikut:

1. Subject oriented (berorientasi subjek)
Data warehouse dirancang untuk membantu user dalam pengambilan

keputusan. Contohnya untuk mengetahui tentang data penjualan perusahaan

Kita bisa membangun data *warehouse* yang berfokus pada penjualan. Dengan menggunakan data *warehouse*, kita bisa menjawab pertanyaan seperti “Siapa pembeli terbaik untuk barang ini tahun lalu?”. Kemampuan untuk mendefinisikan sebuah data *warehouse* sebagai sebuah subjek, dalam hal ini penjualan, membuat data *warehouse* *subject oriented*.

2. *Integrated* (terintegrasi)

Data *warehouse* dikonstruksikan dengan cara mengintegrasikan sejumlah sumber data yang berbeda. Data yang terintegrasi menyebabkan data tersebut lebih konsisten, sehingga lebih mudah dipahami oleh para pembuat keputusan.

3. *Time-variant*

Data *warehouse* harus bisa menghasilkan informasi dari sudut pandang historical (misalnya informasi 5-10 tahun yang lalu atau bahkan lebih). Atau bisa dikatakan bahwa data *warehouse* berfokus pada perubahan setiap waktunya.

4. *Non-volatile*

Data yang ada dalam data *warehouse* tidak bisa di-edit ataupun di-update.

Data *warehouse* dibuat untuk melayani *user* (*analyst* dan pengambil keputusan). Sehingga data *warehouse* wajib dirancang sesuai dengan persyaratan [3] berikut:

1. Harus bisa memberikan kepuasan kepada setiap *user*.
2. Memiliki *function* sendiri tanpa mengganggu OLTP systems.
3. Menyediakan pusat tempat penyimpanan data yang konsisten.
4. Menjawab setiap *complex queries* dengan cepat.
5. Menyediakan berbagai analisis *tools* yang kuat, seperti OLAP dan data *mining*.

Sebagian besar data *warehouse* yang sukses selain memenuhi persyaratan di atas juga memiliki beberapa karakteristik

[1] seperti:

1. Berdasarkan model dimensional.
2. Mengandung *historical data*.
3. Terdiri dari *detailed* dan *summarized data*.
4. Tetap mempertahankan konsistensi data walaupun berasal dari sumber yang berbeda.
5. Fokus dalam *single subject*, seperti penjualan, keuangan, atau inventarisasi.

Data *warehouse* bisa dikatakan sebagai suatu salinan dari OLTP (*On-Line Transaction Processing*) yang terstruktur yang digunakan untuk memenuhi kebutuhan analisis, *reporting*, maupun data *mining* [1]. OLTP sendiri adalah sebuah proses yang menitikberatkan pada transaksinya, seperti input data dan lain-lain. Pada Tabel 1 akan memperlihatkan perbedaan OLTP systems dan data *warehouse*.

Tujuan Data Warehouse

Berikut ini adalah tujuan dari data *warehouse* beserta penjelasannya [2]:

1. Memberikan kemudahan untuk mengakses informasi yang ada. Kemudahan disini berbicara tentang efisiensi. Data *warehouse* harus efisien sehingga dengan mudah dipahami oleh *user* bukan hanya oleh *develeper* saja. Selain itu, pengguna juga dapat mengkombinasikan data dalam data *warehouse* dengan berbagai cara (*slicing and dicing*). Untuk mengakses data *warehouse* disarankan sebaiknya dapat dilakukan dengan sederhana dan mudah dioperasikan.
2. Menyediakan informasi yang konsisten. Data *warehouse* hanya berisi informasi-informasi yang relevan bagi kebutuhan *user* untuk pengambil keputusan. Oleh karena itu, kredibilitas data yang terdapat dalam data *warehouse* harus dapat dipertanggungjawabkan.
3. Mampu beradaptasi dan tahan terhadap

Tabel 1. Perbedaan OLTP dan Data Warehouse

OLTP	Data Warehouse
Dirancang untuk operasi <i>real-time</i> bisnis	Dirancang untuk analisis dari suatu bisnis berdasarkan atribut dan kategori
OLTP	Data Warehouse
Data disimpan pada beberapa <i>platform</i>	Data disimpan pada satu platform saja
Data diorganisir berdasarkan fungsi atau operasinya	Data diorganisir berdasarkan subjek
Prosesnya bersifat berulang (<i>loop</i>)	Prosesnya dilakukan setiap saat dan harus berorientasikan waktu (<i>historical</i>)
Untuk operasional	Untuk managerial
Berorientasi pada transaksi	Berorientasi pada analisis

perubahan.

Perubahan-perubahan yang terjadi harus dapat diatasi oleh data *warehouse*. Dengan kata lain, data *warehouse* harus dirancang agar mampu menghadapi setiap perubahan dengan terencana. Hal ini berarti perubahan yang terjadi tidak boleh merusak atau mengganggu data dan aplikasi yang telah ada sebelumnya.

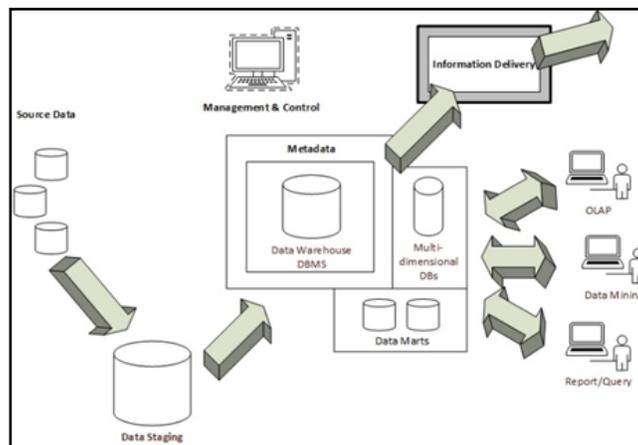
4. Mampu mengamankan informasi.
Informasi yang tersimpan dalam data *warehouse* harus tersimpan dengan aman. Dengan kata lain, informasi tersebut tidak boleh sampai jatuh ke tangan yang salah. Oleh karena itu, data *warehouse* harus mampu mengendalikan setiap akses dari informasi yang ada.
5. Mampu memberikan dukungan dalam pengambilan keputusan.
Ini merupakan tujuan yang paling penting dan harus ada dalam setiap pembuatan data *warehouse*. Data *warehouse* bisa digambarkan sebagai kumpulan teknologi pendukung keputusan, dimaksudkan agar setiap pekerjaan yang berhubungan dengan informasi, dapat membuat keputusan dengan cepat dan tepat.
6. *User friendly*.

Seperti pada tujuan data *warehouse* pertama, data *warehouse* harus dirancang agar dapat dioperasikan dengan mudah oleh *user*. Tidak seperti sistem operasional dimana seringkali *user* tidak memiliki pilihan yang lain kecuali menggunakan sistem baru, akan tetapi *user* data *warehouse* biasanya merupakan pilihan. Oleh karena itu, proses penentuan *user* data *warehouse* merupakan faktor yang sangat penting.

Arsitektur Data Warehouse

Arsitektur data *warehouse* bisa dilihat pada Gambar 1 dikelompokkan menjadi [2]:

1. *Source Data*
Merupakan sumber data, atau bisa dibilang bahwa darimana data itu berasal. Untuk membangun data *warehouse* maka *source* data berasal dari operasional system atau OLTP *database*.
2. *Data Staging*
Merupakan sebuah proses yang diperlukan sebelum data *source* masuk ke dalam data *warehouse*. Proses ini dinamakan proses ETL (*Extracting, Transformation, loading*).
3. *Data Warehouse*



Gambar 1. Arsitektur Data Warehouse

Merupakan tempat penyimpanan data yang multidimensi, dimana data yg tersimpan berupa *metadata*, *summary data* dan *raw data*.

4. Data *Marts*

Merupakan bagian dari data *warehouse*. Seluruh data *mart* jika digabungkan akan menjadi satu data *warehouse*.

5. *Users*

Merupakan pengguna yang akan menggunakan data *warehouse*. *Users* bisa memiliki tugas yang berbeda-beda, ada yang untuk analisis, ada yg untuk *reporting*, dan ada yang untuk membentuk data *mining*. Intinya informasi yang berasal dari data *warehouse* akan diolah menjadi “sesuatu” menggunakan *tools* yang ada yang bisa berguna untuk pengambilan keputusan.

Dimensional Model Data Warehouse

Kebutuhan *user* dan realitas data yang menjadi faktor penentu untuk merancang dimensional model data *warehouse*, seperti bisnis apa yang paling diperlukan, detailnya seperti apa dan dimensi-dimensi serta fakta-fakta apa yang harus diikutkan. [2].

Dimensional model harus sesuai dengan kebutuhan dari *user*. Model ini juga harus dirancang sedemikian rupa sehingga mudah

untuk dipertahankan dan dapat beradaptasi dari segala perubahan yang akan terjadi. Desain modelnya harus dihasilkan dalam bentuk *database* relasional yang mendukung OLAP *cubes* untuk menyediakan secara “instant” hasil *query* untuk analisis.

1. Tabel Dimensi (*Dimension Tables*)

Tabel dimensi menjelaskan tentang entitas bisnis dari suatu enterprise [2]. Tabel dimensi umumnya berisi data keterangan, dimana data tersebut jarang sekali mengalami perubahan.

2. Tabel Fakta (*Fact Tables*)

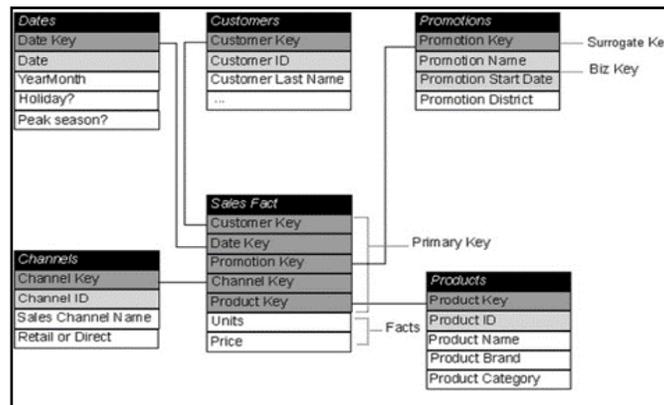
Tabel fakta merupakan sebuah tabel yang menjelaskan tentang transaksi bisnis dari suatu enterprise biasanya disebut tabel detail [2]. Tabel fakta umumnya berisi data yang berkaitan langsung dengan proses bisnisnya.

3. Skema *Dimensional Model*

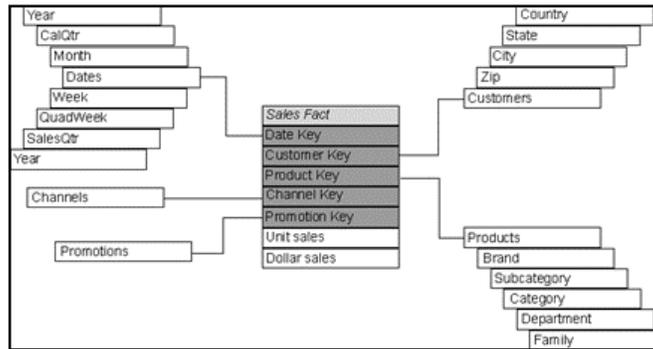
Berikut ini adalah beberapa skema yang biasa digunakan untuk merancang suatu data *warehouse*:

a) Skema *star*

Suatu skema disebut skema *star* jika seluruh tabel dimensi dihubungkan secara langsung ke tabel fakta dan satu tabel fakta wajib memiliki relasi minimal dengan satu tabel dimensi [2]. Berikut ini adalah contoh gambar dari skema *star*.



Gambar 2. Skema *Star*



Gambar 3. Skema Snowflake

b) Skema snowflake

Suatu skema disebut skema *snowflake* jika satu atau lebih tabel dimensi tidak berhubungan langsung dengan tabel fakta tetapi harus berhubungan melalui tabel dimensi lain [2]. Berikut ini adalah contoh gambar dari skema *snowflake*.

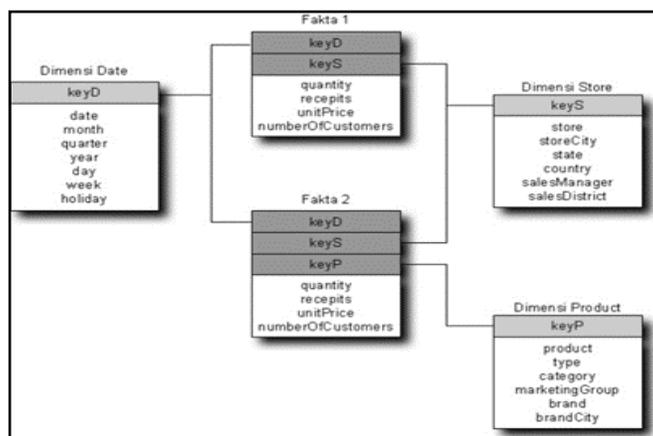
c) Skema Constellation

Suatu skema dikatakan sebagai skema *constellation* jika ada satu tabel dimensi yang dipakai bersamaan oleh satu atau lebih tabel fakta [2]. Berikut ini adalah contoh gambar dari skema *constellation*:

Proses ETL Data Warehouse

Proses ETL atau biasa disebut *Extract, Transform, dan Load* merupakan proses pengubahan data dari OLTP *database* menjadi data *warehouse*. Jika dilihat dari arsitektur data *warehouse*, proses ETL ini merupakan proses yang berada di data *staging*.

Proses ETL merupakan proses untuk mengubah, memformat ulang serta mengintegrasikan data yang berasal dari satu atau beberapa OLTP *systems* [1].



Gambar 4. Skema Constellation

1. Extraction

Extraction merupakan sebuah proses dimana proses tersebut mencari data yang berasal dari OLTP kemudian dengan menggunakan beberapa kriteria yang sudah diberikan untuk memilah data dan juga untuk mencari data yang berkualitas, kemudian data tersebut diangkut ke file lain atau *database*. [1].

2. Transformation

Data *transformation* merupakan suatu fase yang terjadi ketika data sudah menjadi *raw data* (hasil *extraction*) diubah menjadi bentuk yang sudah ditetapkan dimana bentuk tersebut harus bisa digunakan dalam data *warehouse*. Berikut ini adalah beberapa proses dasar yang harus ada dalam data *transformation* :

- a) *Selection*
Memilih atau memilah data hasil dari *extraction*.
- b) *Splitting/Joining*
Splitting/joining meliputi tipe-tipe manipulasi data yang perlu dilakukan pada proses *selection*.
- c) *Conversion*
Proses ini merupakan tahapan paling penting. Pada tahap *conversion*, data hasil *selection* kemudian akan diubah menjadi data yang layak digunakan pada data *warehouse*.
- d) *Summarization*
Tahap ini merupakan tahap pembentukan model yang akan ditampilkan kepada *user*.
- e) *Enrichment*
Tahap ini merupakan tahap pembentukan kembali serta penyederhanaan *field* yang ada untuk membuat *field* tersebut lebih berguna pada data *warehouse*.

3. Loading

Loading adalah suatu proses pemindahan data secara fisik dari OLTP *systems* ke dalam data *warehouse*.

Operasi *loading* terdiri dari memasukkan *record* ke dalam bermacam-macam dimensi dan *fact* tabel dari data *warehouse*.

Data Warehouse Tools

Berikut ini adalah *tools* yang digunakan *users* setelah data *warehouse* terbentuk dengan tujuan yang berbeda-beda [2]:

1. OLAP (On-Line Analytical Processing)

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*. OLAP dirancang untuk beroperasi secara efisien dengan data yang terorganisir sesuai dengan model dimensi umum yang biasa digunakan dalam data *warehouse*.

Tidak ada data *warehouse* modern yang selesai dengan sempurna tanpa fungsionalitas OLAP. Tanpa OLAP, kita tidak dapat memberikan *users* seluruh kemampuan untuk melakukan analisis multidimensional, untuk melihat informasi dari segala sudut pandang, dan untuk membuat keputusan yang bersifat krusial. Oleh karena itu, OLAP sangat krusial. Berikut ini adalah karakteristik paling mendasar dalam OLAP *systems* [1]:

- a) Memberikan para pelaku bisnis pandangan logis yang multidimensi dari data yang ada dalam data *warehouse*.
- b) Memfasilitasi *query* interaktif dan kompleks analisis kepada *users*.
- c) Memungkinkan *user* untuk *drill-down* atau *roll-up* data yang ada baik untuk *single dimension* maupun untuk *multi dimension*.
- d) Memberikan kemampuan untuk melakukan perhitungan dan

perbandingan yang rumit.

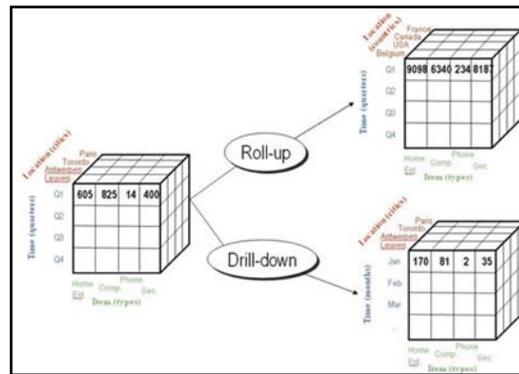
- e) Hasilnya bisa dipresentasikan dalam bentuk yang lebih berarti, seperti grafik atau tabel.

Kegunaan OLAP [1] antara lain:

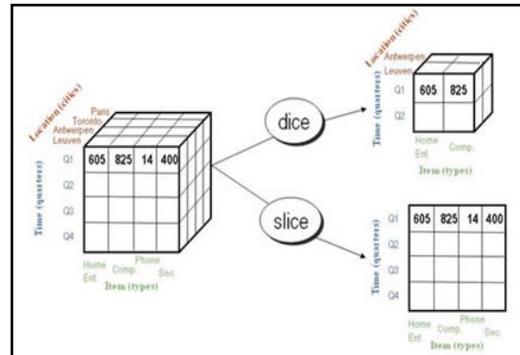
- a) Meningkatkan produktivitas dari bisnis manajer, eksekutif dan analis.
- b) Memanfaatkan OLAP dengan baik bisa membuat *users* bisa dengan percaya diri membuat analisis mereka sendiri tanpa bantuan IT assistance.
- c) Keuntungan untuk IT developers yaitu penggunaan OLAP bisa sangat membantu mempercepat kinerja dari aplikasinya sendiri.
- d) Meningkatkan efisiensi kerja.

OLAP dapat digunakan untuk melakukan hal-hal seperti [1]:

- a) *Consolidation (roll-up)*
Konsolidasi melibatkan pengelompokan data.
- b) *Drill-down*
Suatu bentuk yang merupakan kebalikan dari konsolidasi, untuk mendapatkan lebih detail tentang suatu dimensi serta bisa dikatakan sebagai suatu navigasi dari tingkat yang lebih umum ke tingkat yang lebih spesifik. Untuk lebih jelasnya *roll-up* dan *drill-down* dapat dilihat pada Gambar 5.
- c) *Slicing and dicing*
Menjabarkan pada kemampuan untuk melihat data dari berbagai sudut pandang. Untuk lebih jelasnya *slicing and dicing* bisa dilihat pada Gambar 6.
- d) *Pivot*
Menampilkan nilai-nilai ukuran dalam tata letak tabel yang berbeda dan juga bisa mengatur kembali dimensi dalam OLAP cube. Untuk lebih jelasnya *pivot* bisa dilihat pada Gambar 7.



Gambar 5. Roll-up dan Drill-down

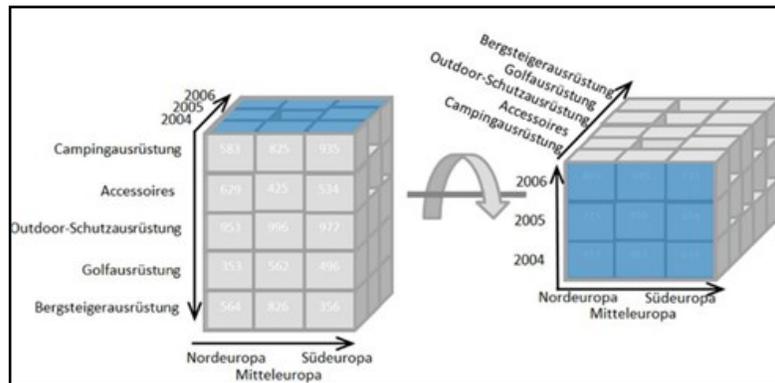


Gambar 6. Slicing and Dicing

Untuk memodelkan OLAP kepada *users*, kita gunakan OLAP cube. OLAP cube adalah bagian utama dari OLAP yang berisi kumpulan data yang banyak dan disatukan (agregasi) untuk mempercepat hasil *query* [1]. OLAP cube contohnya seperti pada Gambar 5, Gambar 6, dan Gambar 7.

2. Reporting

Reporting tools merupakan *tools* yang digunakan untuk mempermudah *user* memperoleh data yang sudah lama ataupun data sekarang dan melakukan beberapa standard analisis statistik. Data yang dihasilkan dari *reporting tools* bisa berupa bentuk laporan biasa dan juga bisa berupa grafik.



Gambar 7. Pivot

3. Data mining

Data *mining* merupakan teknologi yang mengaplikasikan algoritma yang canggih dan kompleks untuk menganalisis data dan mencari informasi yang menarik dari kumpulan data tersebut. Perbedaan mendasar antara OLAP dan data *mining* yaitu terletak pada apa yang akan dianalisisnya. Pada OLAP, yang dianalisis adalah modelnya, tetapi pada data *mining* yang dianalisis adalah datanya (harus berjumlah besar).

TUJUAN PENELITIAN

Tujuan dari penelitian ini adalah sebagai berikut:

“Memudahkan jurusan Teknik Informatika dalam menghimpun kebutuhan informasi strategis yang dibentuk melalui proses rekayasa kebutuhan dari berbagai sumber data (basis data dan file eksternal) “

METODOLOGI PENELITIAN

Jenis Penelitian

Jenis penelitian dalam pelaksanaan penelitian ini adalah penelitian terapan (*applied research, practical research*). Jenis penelitian terapan adalah penyelidikan yang hati-hati, sistematis, dan terus menerus terhadap suatu masalah dengan tujuan

untuk digunakan dengan segera untuk keperluan tertentu. Penelitian terapan memilih masalah yang ada hubungannya dengan keinginan masyarakat serta untuk memperbaiki praktik-praktik yang ada.

Metode Penelitian

Untuk metode perancangan data *warehouse* dibagi menjadi beberapa tahapan, yaitu:

1. Planning

Tahap ini merupakan tahap untuk merencanakan perancangan data *warehouse* di jurusan Teknik Informatika UNIKOM. Perencanaan ini terdiri dari kegiatan:

- a) Pemilihan ketua tim dan anggota tim
- b) Pembagian kerja tim
- c) Pemetaan cakupan permasalahan, waktu, dan biaya
- d) Pembuatan proposal penelitian.

2. Requirement Engineering

Tahap ini merupakan tahap untuk mengumpulkan kebutuhan beserta fakta yang akan digunakan dalam melakukan penelitian ini. Kebutuhan dan fakta dalam penelitian ini didapat dengan cara:

- a) Wawancara
Wawancara pada penelitian ini dilakukan kepada pihak-pihak atau entitas luar dari sistem informasi

yang ada di jurusan Teknik Informatika.

- b) Observasi
Observasi pada penelitian ini dilakukan dengan cara pengamatan dan bekerja langsung di tempat penelitian.

3. Analisis Kebutuhan

Tahap ini merupakan tahap untuk mempelajari dan memahami kebutuhan yang telah dikumpulkan pada tahap *requirement engineering*. Selanjutnya, hasil pemahaman ini digunakan untuk menentukan kebutuhan perancangan data warehouse secara garis besar sebagai tahap persiapan menuju tahap perancangan.

4. Design Architecture

Tahap ini merupakan tahap untuk membentuk arsitektur data warehouse yang didasari hasil dari tahap analisis. Hasil dari perancangan tersebut dapat digunakan untuk mengimplementasikan data warehouse dalam bentuk perangkat lunak.

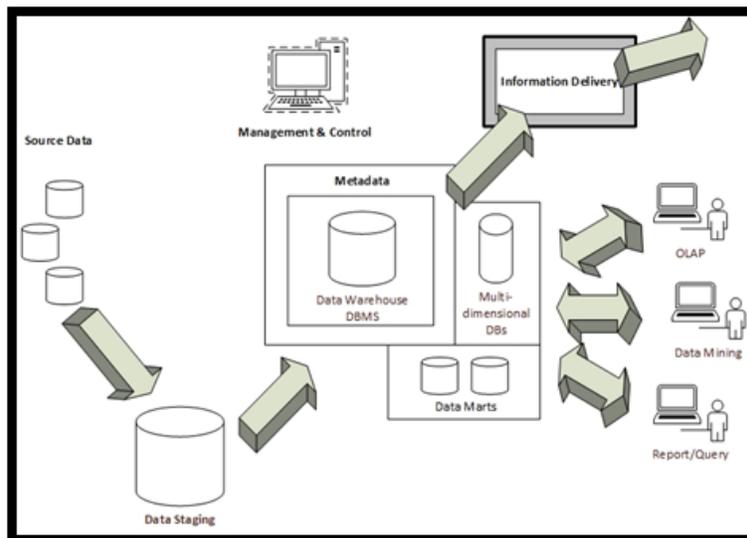
HASIL DAN PEMBAHASAN

Untuk bisa membangun model data warehouse di Program Studi Teknik Informatika UNIKOM maka langkah pertama yang harus dilakukan adalah memahami arsitektur data warehouse yang akan digunakan. Secara umum, arsitektur data warehouse dapat dilihat pada Gambar 8.

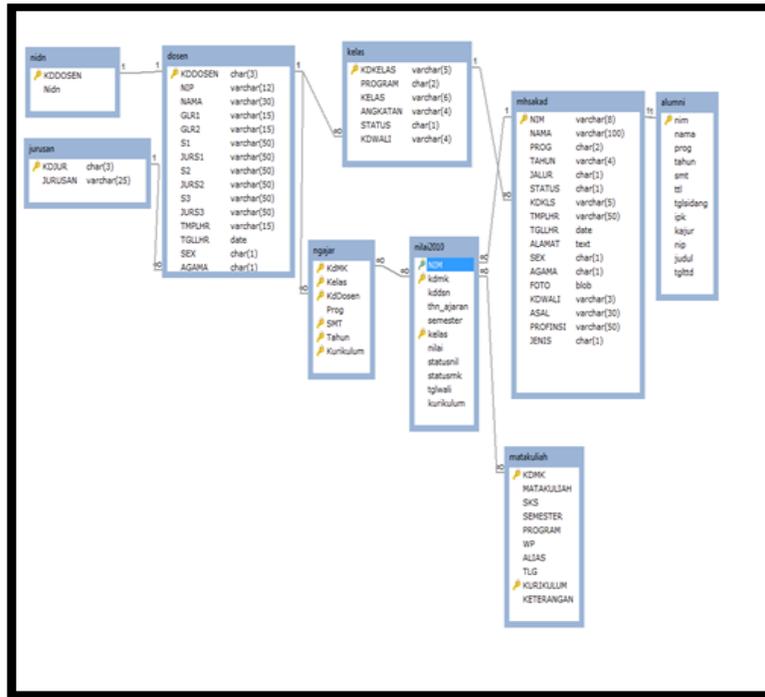
Pada penelitian ini hanya akan dibahas tiga lapisan pada arsitektur data warehouse karena langkah pemodelan data warehouse hanya membutuhkan tiga lapisan awal saja. Lapisan yang akan digunakan pada penelitian ini adalah sebagai berikut:

1. Lapisan *Data Source*

Pada lapisan ini akan dibahas tentang sumber data yang akan digunakan untuk membangun model data warehouse pada penelitian ini. Sumber data yang akan digunakan pada penelitian ini berasal dari basis data Program Studi Teknik Informatika. Format data yang digunakan adalah .sql dengan DBMS MySQL. Tabel-



Gambar 8. Arsitektur Umum Data Warehouse [1]



Gambar 9. Diagram Relasi Sumber Data

tabel pada basis data yang akan digunakan dapat dilihat di diagram relasi pada Gambar 9.

Penjelasan dari masing-masing tabel adalah sebagai berikut:

- a) Tabel Jurusan
Struktur dari tabel jurusan dapat dilihat pada Gambar 10.
- b) Tabel Mata Kuliah
Struktur dari tabel mata kuliah dapat dilihat pada Gambar 11.

Field	Type
KDJUR	char(3) NOT NULL
JURUSAN	varchar(25) NULL

Gambar 10. Struktur Tabel Jurusan

Field	Type
KDMK	varchar(8) NOT NULL
MATAKULIAH	varchar(50) NULL
SKS	tinyint(1) unsigned NULL
SEMESTER	tinyint(1) unsigned NULL
PROGRAM	char(2) NOT NULL
WP	char(1) NOT NULL
ALIAS	varchar(30) NULL
TILG	char(1) NOT NULL
KURIKULUM	varchar(4) NOT NULL
KETERANGAN	varchar(30) NULL

Gambar 11. Struktur Tabel Mata Kuliah

- c) Tabel alumni
Struktur dari tabel alumni dapat dilihat pada Gambar 12.

Field	Type
 nim	varchar(8) NOT NULL
nama	varchar(50) NULL
prog	varchar(10) NULL
tahun	varchar(4) NULL
smt	varchar(6) NULL
ttd	varchar(50) NULL
tglsidang	varchar(50) NULL
ipk	float NULL
kajur	varchar(50) NULL
nip	varchar(20) NULL
judul	varchar(255) NULL
tglttd	varchar(50) NULL

Gambar 12. Struktur Tabel Alumni

- d) Tabel Dosen
Struktur dari tabel dosen dapat dilihat pada Gambar 13..

Field	Type
 KDDOSEN	char(3) NOT NULL
NIP	varchar(12) NULL
NAMA	varchar(30) NULL
GLR1	varchar(15) NULL
GLR2	varchar(15) NULL
S1	varchar(50) NULL
JURS1	varchar(50) NULL
S2	varchar(50) NULL
JURS2	varchar(50) NULL
S3	varchar(50) NULL
JURS3	varchar(50) NULL
TMPLHR	varchar(15) NULL
TGLLHR	date NULL
SEX	char(1) NULL
AGAMA	char(1) NULL
ALAMAT	text NULL
TELP	varchar(40) NULL
KDJAB	char(1) NULL
KDJUR	char(3) NULL
AKADEMIK	char(1) NULL
Foto	blob NULL

Gambar 13. Struktur Tabel Dosen

- e) Tabel Kelas
Struktur dari tabel kelas dapat dilihat pada Gambar 14.

Field	Type
 KDKELAS	varchar(5) NOT NULL
PROGRAM	char(2) NULL
KELAS	varchar(6) NULL
ANGKATAN	varchar(4) NULL
STATUS	char(1) NULL
KDWALI	varchar(4) NULL

Gambar 14. Struktur Tabel Kelas

- f) Tabel Mhsakad
Struktur dari tabel mhsakad dapat dilihat pada Gambar 15.

Field	Type
 NIM	varchar(8) NOT NULL
NAMA	varchar(100) NOT NULL
PROG	char(2) NOT NULL
TAHUN	varchar(4) NOT NULL
JALUR	char(1) NOT NULL
STATUS	char(1) NOT NULL
KDKLS	varchar(5) NOT NULL
TMPLHR	varchar(50) NULL
TGLLHR	date NULL
ALAMAT	text NULL
SEX	char(1) NULL
AGAMA	char(1) NULL
FOTO	blob NULL
KDWALI	varchar(3) NULL
ASAL	varchar(30) NULL
PROFINSI	varchar(50) NULL
JENIS	char(1) NULL

Gambar 15. Struktur Tabel Mhsakad

g) Tabel NIDN

Struktur dari tabel NIDN dapat dilihat pada Gambar 16.

Field	Type
KDDOSEN	varchar(3) NOT NULL
Nidn	varchar(10) NULL

Gambar 16. Struktur Tabel NIDN

h) Tabel Nilai

Struktur dari tabel nilai dapat dilihat pada Gambar 17.

Field	Type
NIM	varchar(8) NOT NULL
kdmk	varchar(8) NOT NULL
kddsn	char(3) NULL
thn_ajaran	varchar(4) NULL
semester	varchar(6) NULL
kelas	varchar(50) NOT NULL
nilai	char(1) NULL
statusnil	char(1) NULL
statusmk	char(1) NULL
tglwali	date NULL
kurikulum	varchar(4) NULL

Gambar 17. Struktur Tabel Nilai

i) Tabel Ngajar

Struktur dari tabel ngajar dapat dilihat pada Gambar 18.

Field	Type
KdMK	varchar(8) NOT NULL
Kelas	varchar(50) NOT NULL
KdDosen	char(3) NOT NULL
Prog	char(2) NULL
SMT	varchar(6) NOT NULL
Tahun	varchar(4) NOT NULL
Kurikulum	varchar(4) NOT NULL

Gambar 18. Struktur Tabel Ngajar

Dari tabel-tabel inilah akan dibentuk sebuah model data warehouse di Program Studi Teknik Informatika.

2. Lapisan *Data Staging*

Pada lapisan *Data Staging*, data yang sudah diidentifikasi pada lapisan *Data Source* akan dipersiapkan agar bisa dibentuk menjadi model data warehouse. Adapun proses yang akan dilakukan pada lapisan *data staging* adalah:

a) *Extract*

Proses ekstraksi pada penelitian ini dilakukan dengan cara mengambil seluruh data dari tabel-tabel yang sudah diidentifikasi. Data diambil dalam bentuk apa adanya tanpa melakukan perubahan baik secara tipe data maupun struktur tabel. Adapun tabel yang diekstraksi adalah:

1. Tabel alumni
2. Tabel dosen
3. Tabel jurusan
4. Tabel kelas
5. Tabel matakuliah
6. Tabel mhsakad
7. Tabel ngajar
8. Tabel nidn
9. Tabel nilai.

b) *Transform*

Pada proses *transform* dilakukan beberapa tahap yang berfungsi untuk mempersiapkan data yang akan digunakan pada pemodelan data warehouse. Sebelum melakukan proses *transform*, pemahaman terhadap kebutuhan informasi strategis yang dibutuhkan sangat diperlukan. Hal ini dilakukan agar data yang dihasilkan pada proses *transform* sesuai dengan kebutuhan penggunaannya. Dari hasil observasi dan studi literatur terhadap borang akreditasi Program Studi Teknik Informatika UNIKOM, ditemukan beberapa informasi strategis

yang dibutuhkan oleh pihak Program Studi. Informasi strategis tersebut adalah

1. Jumlah calon mahasiswa reguler yang ikut seleksi dan lulus seleksi per tahun akademik dan totalnya dari seluruh tahun akademik.
2. Jumlah mahasiswa baru baik reguler non transfer maupun transfer per tahun akademik dan totalnya dari seluruh tahun akademik.
3. Jumlah seluruh mahasiswa per tahun akademik dan totalnya dari seluruh tahun akademik.
4. Jumlah lulusan baru baik reguler non transfer maupun transfer per tahun akademik dan totalnya dari seluruh tahun akademik.
5. IPK mahasiswa reguler yang terbagi menjadi IPK rata-rata, IPK terendah, dan IPK tertinggi.
6. Jumlah persentase lulusan reguler yang terbagi menjadi mahasiswa yang memiliki IPK < 2.75, IPK 2.75-3.50, dan IPK >3.50 per tahun akademik
7. Jumlah mahasiswa yang mendapat beasiswa per tahun akademik dan per jenis beasiswa yang didapat.
8. Jumlah prestasi mahasiswa per tahun akademik dan berdasarkan tingkat dari kompetisinya (local, regional, nasional, dan internasional)
9. Rata-rata waktu tunggu lulusan untuk mendapatkan pekerjaan per bidang pekerjaannya.
10. Persentasi lulusan yang pekerjaannya sesuai dengan bidang ilmunya per tahun akademik.
11. Jumlah mahasiswa yang mendapatkan nilai tertentu (A – E) per tahun akademik dan per mata kuliahnya.
12. Jumlah dosen yang aktif mengajar tiap tahun akademik dan mata kuliahnya.
13. Jumlah dosen yang telah memiliki NIDN.

Pada penelitian ini tahap-tahap pada proses *transform* yang dilakukan dapat dilihat pada 19.



Gambar 19. Tahapan Proses Transform

Penjelasan dari tiap-tiap proses tersebut adalah sebagai berikut:

1. *Selection*

Proses seleksi dilakukan untuk memilih tabel-tabel serta atribut-atribut dari tabel yang akan digunakan pada proses pembentukan model data warehouse. Adapun hasil dari proses seleksi dapat dilihat pada Tabel .

Tabel 2. Hasil Seleksi

No.	Tabel Hasil Seleksi	Atribut Terpilih
1	Alumni	Nim
		Nama
		Prog
		Tahun
		Ipk
2	Dosen	Kddosen
		NIP
		Nama
		Glr1
		Glr2
3	Jurusan	Kdjur
		Jurusan
4	Kelas	Kdkelas
		Program
		Kelas

No	Tabel Hasil Seleksi	Atribut Terpilih
5	Matakuliah	Kdmk
		Matakuliah
		SKS
		Semester
		Program
		Kurikulum
6	Mhsakad	NIM
		Nama
		Prog
		Tahun
7	Ngajar	Kdmk
		Kelas
		Kddosen
		Smt
		Tahun
		Kurikulum
8	nidn	Kddosen
		Nidn
9	Nilai2010	NIM
		Kdmk
		Thn_ajaran
		Semester
		Nilai

2. Splitting/Joining

Proses *splitting/joining* dilakukan untuk menggabungkan atau memisahkan atribut tabel-tabel yang sudah terpilih di tahap seleksi agar lebih mudah dalam proses pengacakan data. Proses ini menghasilkan dua tabel baru yang dapat dilihat pada Tabel 3.

Tabel 3. Hasil Splitting/Joining

No.	Tabel Hasil Splitting/Joining	Atribut
1	Kurikulum	Id_kurikulum
		Thn_kurikulum
2	TahunAjaran	Id_thn
		Thn_ajaran

3. Cleaning

Tahap *cleaning* dilakukan untuk membersihkan data dari *missing value* yang terdapat pada tabel-tabel yang digunakan untuk memodelkan *data warehouse*. Record yang mengandung *Missing value* tidak akan diikutsertakan dalam pembentukan model *data warehouse*.

4. Loading

Setelah melewati tahapan *transform*, tabel-tabel tersebut kemudian dimasukkan ke dalam lingkungan di mana *data warehouse* akan dibentuk dan diakses. Proses ini akan menggunakan alat bantu berupa DBMS yang mendukung format data multidimensional, seperti: Oracle atau MS SQL Server.

5. Lapisan Data warehouse

Pada lapisan *data warehouse*, model *data warehouse* dibentuk melalui informasi strategis yang sudah dikumpulkan. Dari informasi strategis yang dikumpulkan akan dipilih dimensi bisnis yang menjadi sudut pandang pihak eksekutif dalam memandang fakta-fakta bisnis yang ada pada instansinya.

Berdasarkan informasi strategis yang dikumpulkan maka tabel dimensi yang dibentuk pada model *data warehouse* dapat dilihat pada Tabel 4.

Tabel 4. Tabel Dimensi yang Terbentuk

No.	Tabel Dimensi	Field
1	Calon Mahasiswa	No_Peserta
		Nama Peserta
		TTL_Peserta
2	Propinsi	Id_propinsi
		Nama_Propinsi
3	Tahun Akademik	Id_Thn_Akademik
		Thn_Akademik
4	Mahasiswa	NIM
		Nama_Mahasiswa
		Tanggal_Masuk
		Tanggal_Lulus
		Status_Mahasiswa .Id_Status
5	Status_Mahasiswa	Id_Status
		Status_Mhs
6	Mata_Kuliah	Kdmk
		Nama_Mk
7	Beasiswa	Kd_Beasiswa
		Jenis_Beasiswa
8	Prestasi	Kd_Prestasi
		Nama_Prestasi
		Tingkat_Kompetisi
9	Pekerjaan	Kd_Pekerjaan
		Jenis_Pekerjaan
10	Dosen	NIP
		Nama_Dosen
11	NIDN	NIP
		NIDN

Berdasarkan tabel dimensi yang terbentuk, dibentuklah tabel fakta yang berisi fakta-fakta yang ingin dianalisis berdasarkan dimensi bisnis yang sudah dibentuk. Tabel fakta yang dibentuk berdasarkan informasi strategis dapat dilihat pada Tabel 5.

Untuk menggambarkan keterhubungan antara tabel dimensi dan fakta digunakan diagram relasi *data warehouse* yang dapat dilihat pada Gambar 20.

Berdasarkan hasil perbandingan antara diagram relasi OLTP dengan diagram relasi *data warehouse* maka ditemukan fakta bahwa atribut data pada diagram relasi OLTP masih belum bisa memenuhi kebutuhan data yang ada pada diagram relasi *data warehouse*. Hal ini disebabkan karena tidak terintegrasinya kebutuhan data eksternal dengan data internal yang ada di dalam Program Studi Teknik Informatika UNIKOM.

KESIMPULAN DAN SARAN

Kesimpulan

Kesimpulan yang dapat diambil pada penelitian ini adalah:

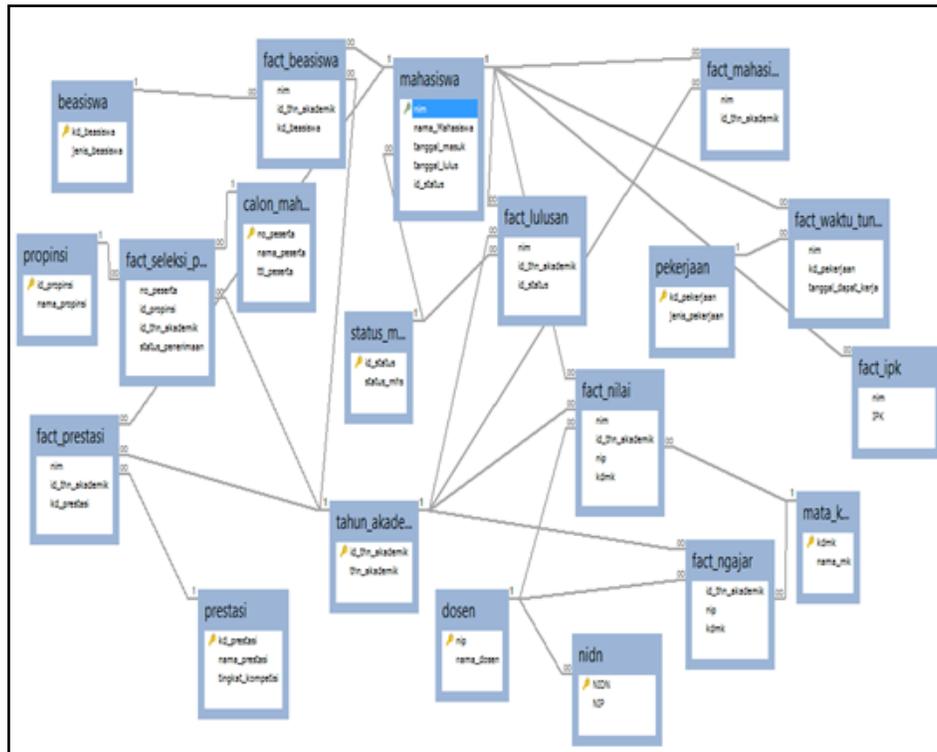
1. Dari kebutuhan informasi strategis yang dibentuk melalui proses rekayasa kebutuhan didapat 11 tabel dimensi dan 9 tabel fakta. T
2. Terdapat kekurangan data pada OLTP Program Studi Teknik Informatika UNIKOM yang mengakibatkan data pada *data warehouse* tidak bisa diisi.

Saran

Saran untuk pengembangan penelitian ini adalah dibutuhkannya integrasi data pada seluruh basis data yang berhubungan dengan *data warehouse* agar model *data warehouse* yang telah dibentuk dapat diimplementasi pada penelitian berikutnya.

Tabel 5. Tabel Fakta yang Terbentuk

No.	Tabel Dimensi	Field
1	Fakta Seleksi PMB	Calon_Mahasiswa.No_Peserta
		Propinsi.Id_Propinsi
		Tahun_Akademik.Id_Thn_Akademik
		Status_Penerimaan
2	Fakta Mahasiswa Aktif	Mahasiswa.NIM
		Tahun_Akademik.Id_Thn_Akademik
3	Fakta Lulusan	Mahasiswa.NIM
		Tahun_Akademik.Id_Thn_Akademik
		Status_Mahasiswa.Id_Status
4	Fakta IPK	Mahasiswa.NIM
		IPK
5	Fakta Beasiswa	Mahasiswa.NIM.
		Tahun_Akademik.Id_Thn_Akademik
		Beasiswa.Kd_Beasiswa
6	Fakta Prestasi	Mahasiswa.NIM.
		Tahun_Akademik.Id_Thn_Akademik
		Prestasi.Kd_Prestasi
7	Fakta Waktu Tunggu Kerja	Mahasiswa.NIM
		Pekerjaan.Kd_Pekerjaan
		Tanggal_Dapat_Kerja
8	Fakta Nilai	Mahasiswa.NIM
		Tahun_Akademik.Id_Thn_Akademik
		Mata_Kuliah.Kdmk
		Dosen.NIP
9	Fakta Dosen Ngajar	Tahun_Akademik.Id_Thn_Akademik
		Mata_Kuliah.Kdmk
		Dosen.NIP



Gambar 20. Diagram Relasi Data Warehouse

DAFTAR PUSTAKA

- W. H. Inmon, Building the Data Warehouse Third Edition, Canada: John Wiley & Sons, Inc., 2002
- P. Ponniah, Data Warehousing Fundamentals. A Comprehensif Guide For IT Professionals, John Wiley & Sons, Inc., 2001