

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Pembuatan *data warehouse* telah banyak dilakukan oleh perusahaan-perusahaan industri yang berorientasi *profit*. *Data warehouse* diharapkan mampu mendukung dan meningkatkan akurasi pengambilan keputusan. Pembuatan *data warehouse* tidak hanya terbatas pada lingkup industri, dalam bidang pendidikan pun sudah banyak yang menerapkan *data warehouse*.

Penelitian mengenai perancangan *data warehouse* sistem informasi eksekutif pernah dilakukan oleh Prasetyo (2011). *Data warehouse* tersebut digunakan untuk mendukung para eksekutif melakukan analisis dalam mendukung pengambilan keputusan strategis. Pemodelan yang digunakan dalam penelitian ini adalah *Star Schema*. Pemilihan *star schema* dikarenakan model ini memungkinkan proses *query* yang diterapkan akan lebih ringan serta mempermudah eksplorasi terhadap data dimensinya. Proses ETL pada penelitian ini menggunakan aplikasi yang bernama *Pentaho Data Integration (PDI)*.

Fitriatun (2015) telah melakukan penelitian tentang perancangan *data warehouse* yang bertujuan untuk menganalisis data hasil studi akademik di Fakultas Teknik Universitas Muhammadiyah Yogyakarta (FT UMY). Penelitian ini menggunakan *SQL Server Management Studio (SSMS)* pada *Microsoft SQL Server*

2012 dan *Integration Service* pada *SQL Server Data Tool* (SSDT) sebagai alat pembangunan *data warehouse*. *Data warehouse* yang dibangun telah dianalisis secara *multidimensional* dengan menggunakan *Analysis Services* Hasil analisis penelitian ditampilkan menggunakan *pivot table* dan *pivot chart* pada *Microsoft Office Excel* 2013.

Pemanfaat *data warehouse* sebagai sarana penunjang penyusunan borang akreditasi standar 3 pada Fakultas Teknologi Informasi Universitas Budi Luhur juga telah dilakukan oleh Windarto (2011). Pemodelan data yang digunakan dalam pengembangan *data warehouse* adalah *star schema*. Arsitektur yang digunakan adalah *independent data mart*. Pada arsitektur ini data tidak disimpan dalam *data warehouse* yang besar, melainkan disimpan dalam beberapa *data mart*.

Hasil penelitian yang telah diuraikan tersebut memiliki penerapan pangkalan data dengan cara yang beragam. Maka pada pembuatan pangkalan data ini terdapat hal yang serupa maupun berbeda dari hasil penelitian sebelumnya. Penelitian ini menerapkan *star schema* sebagai pemodelan data. Sedangkan arsitektur yang digunakan adalah *Operational Data Store and Dimensional Data Store* atau yang dikenal dengan ODS + DDS (Rainardi,2008). Proses ETL dan pengujian dalam pembuatan pangkalan data menggunakan dua *tools* yang tersedia pada yang *SQL Server Data Tool* (SSDT) yaitu *Integration Services* atau yang dikenal *SQL Server Integration Services* (SSIS) dan *Analysis Services* atau yang dikenal *SQL Server Analysis Services* (SSAS).

2.2. Landasan Teori

2.2.1. Akreditasi

Menurut Komite Akreditasi Nasional (KAN), akreditasi adalah pengakuan formal yang diberikan oleh badan akreditasi terhadap kompetensi suatu lembaga atau organisasi dalam melakukan kegiatan penilaian kesesuaian tertentu.

Menurut Badan Akreditasi Nasional (BAN) pada buku II pedoman penyusunan borang, akreditasi adalah proses penilaian terhadap institusi secara keseluruhan untuk mengetahui komitmen institusi terhadap kapasitas institusi dan efektivitas pendidikan, yang didasarkan pada standar akreditasi yang telah ditetapkan.

2.2.2. Borang Akreditasi

Menurut BAN, borang akreditasi adalah alat yang berbentuk dokumen yang digunakan untuk mengumpulkan dan mengungkapkan data dan informasi yang digunakan untuk menilai kelayakan dan mutu institusi perguruan tinggi.

2.2.3. Standar Akreditasi

Akreditasi memiliki standar yang harus dipenuhi oleh institusi perguruan tinggi. Standar akreditasi digunakan untuk mengukur dan menentukan kelayakan dan mutu suatu institusi perguruan tinggi. Terdapat beberapa indikator kunci yang digunakan sebagai dasar pengukuran dan penetapan mutu dan kelayakan kinerja perguruan tinggi. Standar akreditasi perguruan tinggi terdiri dari tujuh buah standar, yaitu :

Standar 1. Visi, misi, tujuan dan sasaran, serta strategi pencapaian

Standar 2. Tata pamong, kepemimpinan, sistem pengelolaan, dan penjaminan mutu

Standar 3. Mahasiswa dan lulusan

Standar 4. Sumber daya manusia

Standar 5. Kurikulum, pembelajaran, dan suasana akademik

Standar 6. Pembiayaan, sarana dan prasarana, serta sistem informasi

Standar 7. Penelitian, pelayanan atau pengabdian kepada masyarakat, dan kerjasama

2.2.4. Data Warehouse

Data warehouse atau pangkalan data merupakan sistem yang memiliki data dari hasil integrasi berbagai sumber data operasional. Menurut Rainardi (2008) pangkalan data merupakan suatu sistem yang memperoleh dan menggabungkan data secara berkala dari berbagai sumber ke dalam penyimpanan data dimensi atau *normalized*.

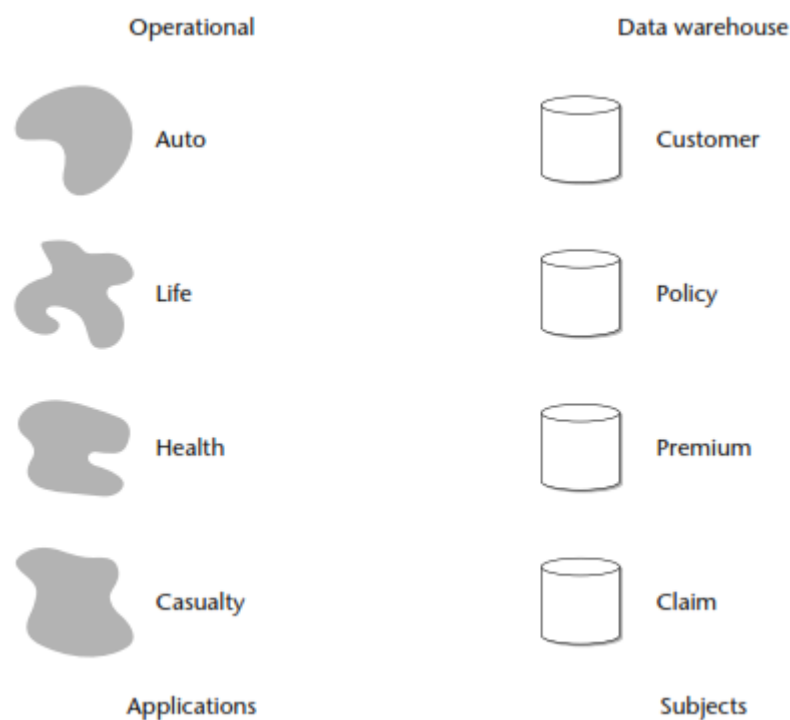
Menurut Kimball dan Caserta (2004) pangkalan data adalah suatu sistem yang mengekstrak, membersihkan, menyesuaikan dan mengirimkan data sumber ke dalam penyimpanan data dimensi dan kemudian mendukung dan menerapkan pencarian dan analisis untuk tujuan pengambilan keputusan.

Pangkalan data terdiri dari satu ataupun lebih *data mart*. *Data mart* merupakan sebuah gabungan tabel fakta dan tabel dimensi yang mengandung pengukuran untuk suatu peristiwa bisnis (Fitriatun, 2015). Data yang terkandung pada pangkalan data dapat dimanfaatkan lebih jauh dengan menggunakan alat seperti *Online Analytical Processing (OLAP)* dan pelaporan (*reporting*).

Menurut Inmon (2005), pangkalan data terdiri dari empat karakteristik utama, yaitu :

a. Berorientasi subject

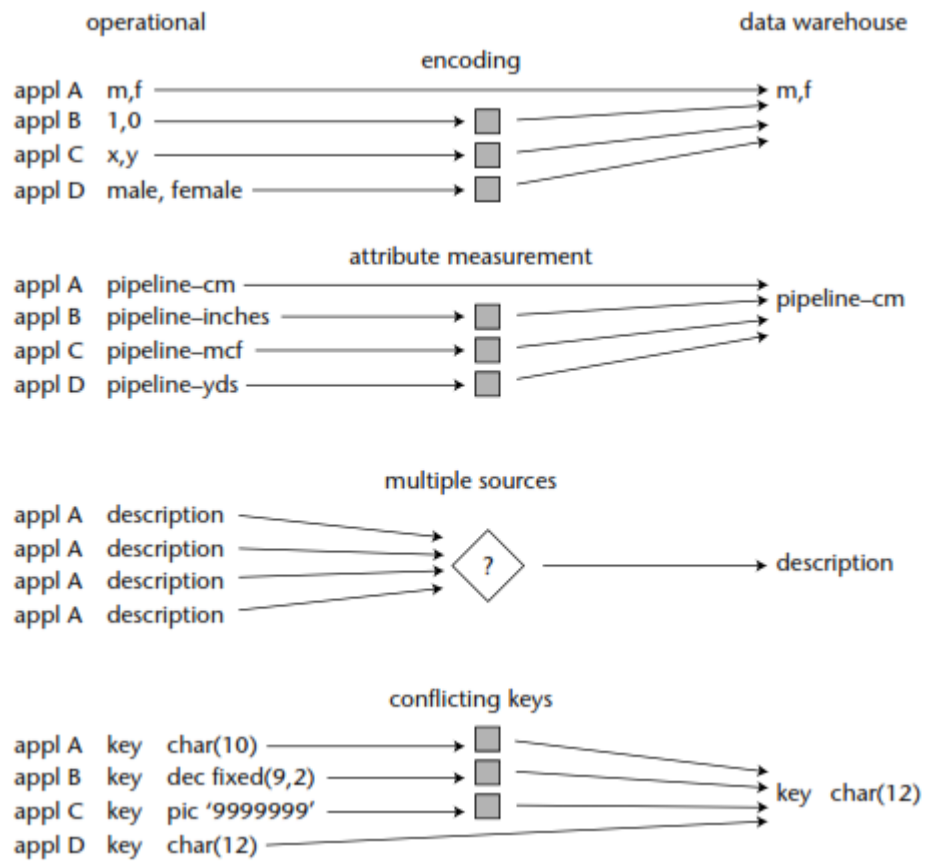
Maksud dari pangkalan data memiliki karakteristik berorientasi subject dikarenakan pangkalan data merupakan sistem yang dirancang untuk menganalisa data berdasarkan subject tertentu, bukan berdasarkan operasional tertentu.



Gambar 1. Berorientasi subjek (Inmon, 2005)

b. Integrasi

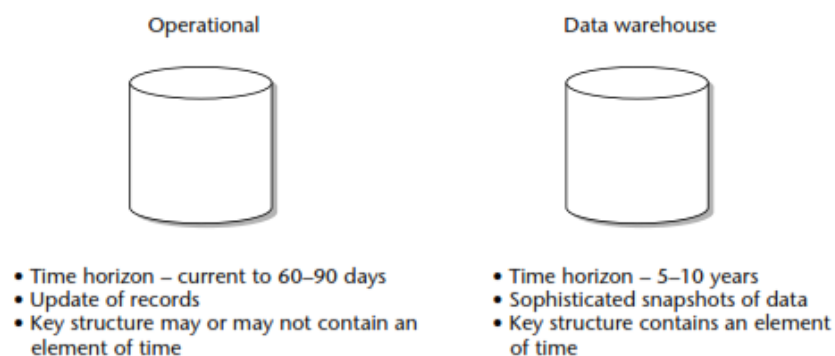
Pangkalan data berkarakter integrasi dikarenakan data yang ada pada sistem pangkalan data merupakan penggabungan data dari beberapa sumber sistem. Data yang berasal dari beberapa sumber akan diubah menjadi format yang konsisten dan saling terintegrasi.



Gambar 2. Integrasi (Inmon, 2005)

c. *Time variant*

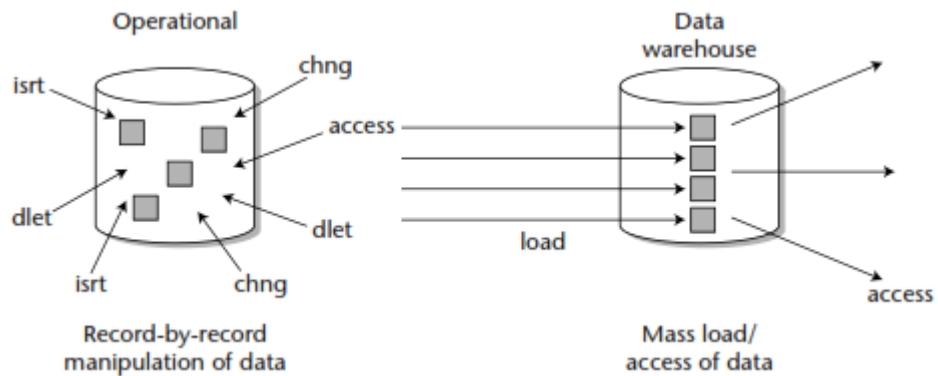
Time variant berarti pangkalan data tidak hanya mengandung data yang sekarang atau terkini, namun juga mengandung data *history*. Data *history* dapat berupa harian, mingguan, bulanan, maupun tahunan.



Gambar 3. *Time variant* (Inmon, 2005)

d. *Nonvolatile*

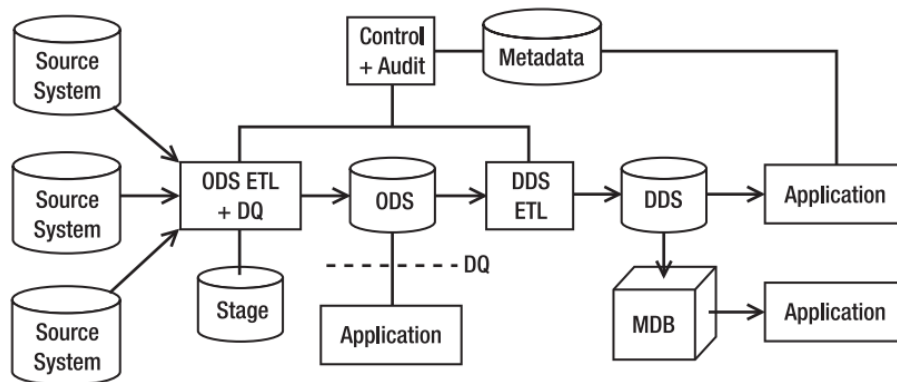
Nonvolatile adalah data yang tersimpan pada pangkalan data bersifat tetap. Maksud bersifat tetap adalah data tidak diperbarui secara *real time* namun akan diperbaharui dalam jangka waktu yang telah ditentukan.



Gambar 4. *Nonvolatile* (Inmon, 2005)

2.2.5. Arsitektur pangkalan data

Menurut Rainadi (2008) arsitektur pangkalan data merupakan suatu hal yang perlu ditentukan ketika membangun sistem pangkalan. Arsitektur pangkalan data menunjukkan bagaimana aliran data dari sumber hingga menjadi pangkalan data. Terdapat beberapa jenis arsitektur pangkalan data yang dapat digunakan dalam pembuatan pangkalan data. Salah satunya yaitu arsitektur *Operational Data Store and Dimensional Data Store* (ODS + DDS) seperti pada Gambar 5.



Gambar 5. Arsitektur ODS + DDS (Rainardi, 2008)

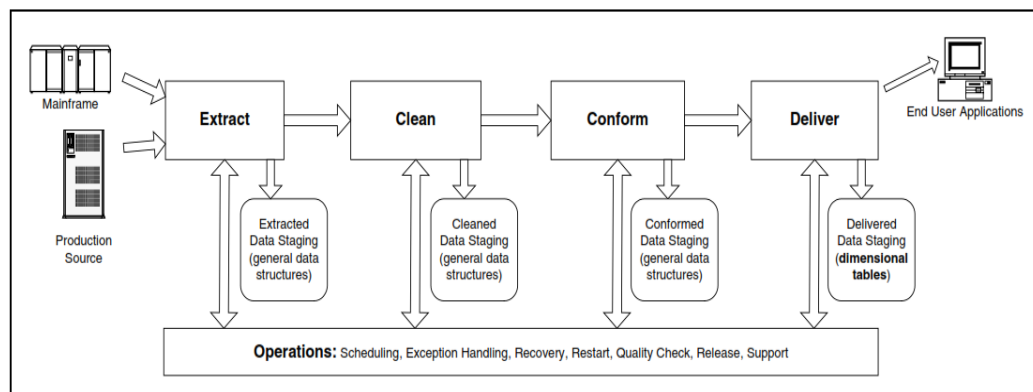
Arsitektur ini memiliki tiga penyimpanan data, yaitu *Stage*, ODS, dan DDS. Penyimpanan data pertama adalah *stage*. *Stage* merupakan tempat penyimpanan data yang bertujuan mempersiapkan data sebelum dimuat ke dalam pangkalan data sehingga data yang dimuat ke dalam ODS merupakan data yang bersih dan telah terintegrasi. Data diambil dari *source system* yang berupa data operasional maupun eksternal data. Penyimpanan data kedua adalah ODS, ODS merupakan penyimpanan data yang dapat di akses oleh *end-user* dan juga merupakan basis data ternormalisasi dalam bentuk normalisasi ketiga atau lebih tinggi. ODS mengandung tabel transaksional dan tabel master. Penyimpanan data ketiga adalah DDS, DDS merupakan *dimensional data store*. Tabel yang ada pada DDS biasanya sudah didenormalisasi. DDS memiliki tabel *dimension* yang berasal dari master tabel dengan penambahan *surrogate key* dan tabel *fact* yang biasanya berasal dari tabel transaksional. Pada arsitektur ODS + DDS hanya memiliki satu tabel *fact*.

2.2.6. *Extract, Transform, dan Load (ETL)*

ETL merupakan sistem yang memiliki kemampuan untuk membaca data dari satu penyimpanan data, merubah data, dan memuat data ke dalam penyimpanan data lain (Rainardi, 2008).

Menurut Connloy dan Begg (2005) pada proses ETL, data harus di rekonstruksi untuk tujuan pangkalan data. Rekonstruksi mencakup hal-hal seperti membersihkan data kotor, menambah atau menghapus kolom, dan denormalisasi data.

Menurut Kimball & Caserta (2004) dalam suatu proses ETL terdiri dari empat tahap, yaitu:



Gambar 6. Empat langkah proses ETL (Kimball dan Caserta,2004)

1. *Extracting.*

Data yang diambil dari sumber sistem biasanya diarahkan langsung ke penyimpanan fisik dengan seminimal mungkin restruktur. Data yang berasal dari sumber sistem terstruktur seperti basis data, biasanya akan diubah menjadi *flat file*. Dengan cara seperti ini memungkinkan proses ekstrak menjadi lebih sederhana, cepat dan fleksibilitas pengulangan proses ekstrak jika ada gangguan.

2. *Cleansing*

Cleansing memiliki tujuan untuk menjadikan data berkualitas sesuai kebutuhan *data warehouse*. Adapun langkah-langkah yang termasuk dalam proses *clean*, seperti memvalidasi nilai, menjamin konsistensi nilai, menghapus nilai duplikat

3. *Conform*

Conform dibutuhkan ketika ada dua atau lebih sumber digabung sebelum dimuat ke dalam *data warehouse*.

4. *Deliver*

Inti dari proses ETL adalah untuk membuat dan memastikan data siap digunakan untuk proses *query*. *Deliver* memasukkan data yang sudah diproses sebelumnya ke dalam *data warehouse*.

2.2.7. Model Dimensional

Model dimensional bertujuan untuk menyajikan data dalam bentuk yang lebih sederhana, mudah dipahami dan memiliki performa yang tinggi. Model dimensional berfokus pada data numerik, seperti jumlah dan rata-rata. Model dimensional memiliki beberapa konsep dasar:

a. *Fact*

Fact merupakan tabel yang mengandung suatu nilai yang dapat diukur atau dapat diperhitungkan seperti jumlah penjualan dan jumlah pembelian. Tabel *fact* terdiri dari kumpulan *foreign key* yang merupakan *primary key* pada masing-masing *dimension*.

b. *Dimension*

Dimension merupakan tabel yang mengandung berbagai atribut yang menjelaskan *foreign key* di tabel *fact*. Tabel *dimension* berisi informasi yang menjelaskan satuan bisnis, seperti pelanggan dan produk.

c. *Measures*

Measures merupakan atribut angka dari *fact*. *Measures* menampilkan nilai berupa angka yang dapat digunakan untuk kepentingan bisnis.

2.2.8. Matriks bus

Sebelum memulai pembuatan pangkalan data, terdapat hal yang perlu dilakukan yaitu mengidentifikasi proses bisnis. Identifikasi proses bisnis dapat dilakukan dengan menggunakan bus matriks. Adapun contoh dari matriks bus dapat dilihat pada Gambar 7. Menurut Kimball dan Ross (2002), matriks bus digunakan untuk membuat, mendokumentasikan, dan mengkomunikasikan arsitektur data.

BUSINESS PROCESSES	COMMON DIMENSIONS							
	<i>Date</i>	<i>Product</i>	<i>Store</i>	<i>Promotion</i>	<i>Warehouse</i>	<i>Vendor</i>	<i>Contract</i>	<i>Shipper</i>
Retail Sales	X	X	X	X				
Retail Inventory	X	X	X					
Retail Deliveries	X	X	X					
Warehouse Inventory	X	X			X	X		
Warehouse Deliveries	X	X			X	X		
Purchase Orders	X	X			X	X	X	X

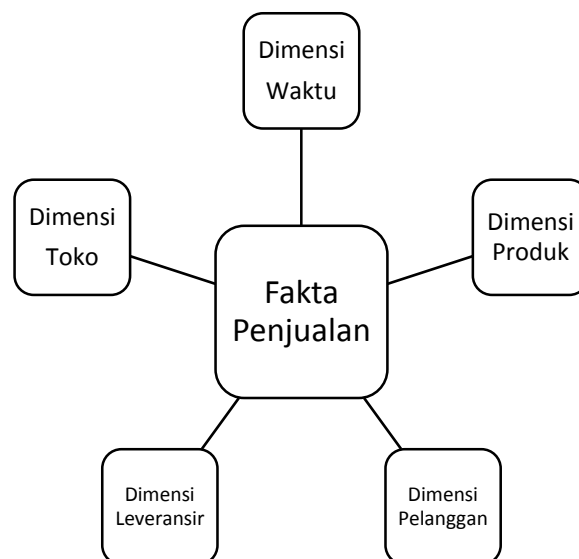
Gambar 7. Matriks bus (Kimball and Ross, 2002)

Baris merupakan acuan dalam mengidentifikasi proses bisnis dan kolom menjelaskan *dimension* yang sesuai dengan proses bisnis. Matriks bus mendefinisikan seperti apa rancangan pangkalan data dengan bentuk yang sederhana.

2.2.9. *Star Schema*

Star schema merupakan model dasar atau model yang paling sederhana yang dapat digunakan dalam model dimensional. Menurut Kimball & Caserta (2004), *Star schema* merupakan skema dasar dalam model dimensional yang menjadi keperluan dasar untuk membangun *Online Analytical Processing* (OLAP) dan secara signifikan mengurangi waktu dalam proses *query*.

Star schema terdiri satu tabel *fact* yang merupakan tabel pusat dan terhubung dengan beberapa tabel kecil yaitu tabel *dimensions*. *Star schema* dapat dilihat pada Gambar 8.

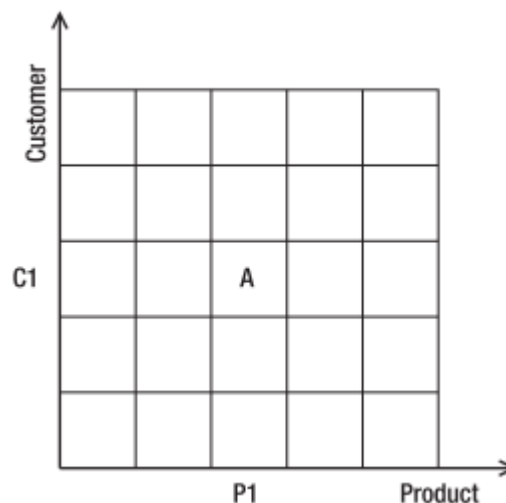


Gambar 8. Star Schema

Pada *star schema*, seluruh *natural key* diganti dengan *surrogate key* sehingga relasi antara tabel *fact* dan tabel *dimension* berdasarkan *surrogate key*. *Star schema* dapat digunakan untuk meningkatkan performa *query* dengan referensi informasi yang telah didenormalisasi ke dalam tabel *single dimension* (Connolly dan Begg, 2005). Denormalisasi dianggap sudah tepat ketika terdapat beberapa tabel *dimension* yang terhubung dengan tabel *dimension* lainnya dan sering digunakan, sehingga menghindari kelebihan penggunaan *join*.

2.2.10. Multidimensional Database (MDB)

Multidimensional Database (MDB) merupakan bentuk basis data dimana data tersimpan dalam suatu *cell* dan posisi tiap-tiap *cell* dijelaskan oleh sejumlah tingkatan yang disebut *dimension* (Rainardi, 2008). Setiap *cell* menjelaskan suatu peristiwa bisnis. Adapun contoh dari MDB pada Gambar 9.



Gambar 9. *Multidimensional database* (Rainardi, 2008)

MDB biasanya digunakan untuk keperluan *online analytical processing* (OLAP). Menurut Rainardi (2008), keuntungan dari penggunaan MDB untuk

OLAP dibanding *relational database* seperti DDS adalah mengurangi penggunaan disk dan memiliki performa yang lebih baik.

2.2.11. Online Analytical Processing (OLAP)

Online Analytical Processing (OLAP) merupakan proses analisis data yang dapat dimanfaatkan untuk kepentingan bisnis. OLAP dapat diterapkan pada *multidimensional database* atau yang dikenal sebagai *multidimensional online analytical processing* (MOLAP).

2.2.12. Microsoft SQL Server 2014

SQL Server merupakan *Relational Database Management System* (RDMS) yang menghubungkan pengguna dengan data untuk pengelolaan basis data. *SQL Server* dapat digunakan untuk melakukan satu ataupun beberapa *server*. Bahasa basis data yang digunakan *SQL Server* adalah *Transact-SQL*.

Transact-SQL merupakan bahasa SQL yang dimiliki oleh *SQL Server* yang berguna bagi pengguna untuk mendapatkan satu atau kumpulan data pada basis data dengan cara menjalankan perintah dari suatu pernyataan SQL. Ada beberapa contoh dari bahasa *transact-SQL*, sebagai berikut:

a. Data Definition Language (DDL)

DDL merupakan kosa kata yang digunakan untuk membangun struktur dari suatu basis data. Penerapan DDL ditujukan kepada objek-objek seperti basis data, tabel dan kolom. Adapun beberapa contoh dari DDL beserta fungsinya pada Tabel 1.

Tabel 1. *Data Definition Language*

<i>Statement</i>	Fungsi
CREATE	Membuat objek baru
ALTER	Mengubah objek yang ada
DROP	Menghapus objek

b. Data Manipulation Language (DML)

DML merupakan kosa kata yang digunakan untuk memanipulasi data dari suatu basis data. Adapun beberapa contoh dari DML beserta fungsinya dapat dilihat pada Tabel 2.

Tabel 2. *Data Manipulation Language*

<i>Statement</i>	Fungsi
SELECT	Mengambil data
INSERT	Memasukkan data
DELETE	Menghapus data
UPDATE	Mengubah data

Ada beberapa sintak yang biasa digunakan melakukan pengambilan, penghapusan dan perubahan data secara spesifik atau berdasarkan suatu ketentuan, seperti pada Tabel 3.

Tabel 3. Sintak pencarian spesifik data

Sintak	Fungsi
WHERE	Mencari data dengan kondisi yang spesifik
GROUP BY	Mengambil data secara pengelompokan berdasarkan satu kolom atau lebih.

(Lanjutan) Tabel 3. Sintak pencarian spesifik data

Sintak	Fungsi
ORDER BY	Menentukan urutan hasil berdasarkan kolom
PARTITION BY	Membagi hasil <i>query</i> berdasarkan partisi.
HAVING	Mencari data dengan kondisi spesifik untuk kelompok dan hanya digunakan dengan SELECT.

c. *Aggregate Functions*

Aggregate Functions merupakan fungsi kalkulasi yang menghasilkan suatu nilai. Beberapa sintak yang ada pada *aggregate functions* ditunjukkan pada Tabel 4.

Tabel 4. *Aggregate Functions*

Sintak	Fungsi
SUM	Mendapatkan total jumlah dari suatu kolom numeric
COUNT	Mendapatkan jumlah data berdasarkan baris
AVG	Mendapatkan nilai rata-rata dari suatu kolom numerik
MAX	Mendapatkan nilai maksimal dari suatu kolom
MIN	Mendapatkan nilai minimum dari suatu kolom

d. *Join Operation*

Salah satu jenis *join operation* adalah OUTER JOIN. OUTER JOIN berfungsi untuk menggabungkan dua atau lebih tabel menjadi satu tabel dan tetap mempertahankan nilai *null* pada hasil pencarian. Dalam

penggabungan tabel biasanya ditemukan nilai suatu kolom yang tidak sesuai dengan nilai kolom pada tabel lainnya. Kondisi seperti inilah yang memberikan nilai *null*. Selain OUTER JOIN, di dalam *join operation* juga terdapat INNER JOIN. INNER JOIN hanya menghasilkan nilai yang *match* di masing-masing tabel. Adapun *join operation* ditampilkan pada Tabel 5.

Tabel 5. *Join operation*

Sintak	Fungsi
LEFT OUTER JOIN	Menggabungkan tabel dan mengembalikan hasil data sejumlah dengan data yang ada pada tabel sebelah kiri.
RIGHT OUTER JOIN	Menggabungkan tabel dan mengembalikan hasil data sejumlah dengan data yang ada pada tabel sebelah kanan.
FULL OUTER JOIN	Menggabungkan tabel dan mengembalikan hasil data sejumlah dengan data yang ada pada tabel sebelah kiri dan kanan serta tetap mempertahankan nilai yang <i>match</i> maupun <i>no match</i> .

2.2.13. *SQL Server Data Tool (SSDT) 2013*

SSDT merupakan alat pengembangan basis data modern. SSDT disediakan untuk lingkungan pengembangan *business intelligence* yang di dalamnya juga mencakup *SQL Server Integration Services* dan *SQL Server Analysis Services*.

a. *SQL Server Integration Services (SSIS)*

SQL Server Integration Services (SSIS) merupakan suatu program yang bertujuan sebagai solusi integrasi data. Program ini memiliki kemampuan untuk mengekstrak dan memuat data dari beragam sumber dan ke beragam tujuan data.

b. *SQL Server Analysis Services (SSAS)*

SQL Server Analysis Services (SSAS) merupakan *tool* yang digunakan untuk melakukan *Online Analytical Processing (OLAP)* serta berguna untuk melakukan proses *query* secara *multidimensional* dan kalkulasi dalam lingkungan *business intelligence*.