



**SINKRONISASI BASIS DATA OPERASIONAL  
MENJADI INDEPENDENT DATA MART  
STUDI KASUS: PT MANE INDONESIA**

**ELFRIDA SURYANI LUMBANTOBING  
41505120092**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS MERCU BUANA  
JAKARTA  
2009**

## **LEMBAR PERNYATAAN**

Yang bertanda tangan dibawah ini:

NIM : 41505120092  
Nama : ELFRIDA SURYANI LUMBANTOBING  
Judul Skripsi : SINKRONISASI BASIS DATA OPERASIONAL  
MENJADI INDEPENDENT DATA MART  
STUDI KASUS: PT MANE INDONESIA

Menyatakan bahwa skripsi tersebut diatas adalah hasil karya saya sendiri dan bukan plagiat. Apabila ternyata ditemukan didalam laporan skripsi saya terdapat unsur plagiat, maka saya siap untuk mendapatkan sanksi akademik yang terkait dengan hal tersebut.

Jakarta, 27 Juni 2009

(Elfrida Suryani Lumbantobing)

## LEMBAR PERSETUJUAN

NIM : 41505120092  
Nama : ELFRIDA SURYANI LUMBANTOBING  
Judul Skripsi : SINKRONISASI BASIS DATA OPERASIONAL  
MENJADI INDEPENDENT DATA MART  
STUDI KASUS: PT MANE INDONESIA

SKRIPSI INI TELAH DIPERIKSA DAN DISETUJUI

JAKARTA, 27 JUNI 2009

Devi Fitrianah, S.Kom., MTI  
Pembimbing

Devi Fitrianah, S.Kom., MTI  
Koord. Tugas Akhir Teknik Informatika

Abdusy Syarif, ST., MT  
KaProdi Teknik Informatika

## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat dan karunia-Nya sehingga tugas akhir yang disusun sebagai salah satu syarat kelulusan sarjana strata satu (S1) di program studi Teknik Informatika Universitas Mercu Buana ini dapat diselesaikan dengan baik.

Menyertai selesainya tugas akhir ini, terimakasih diucapkan kepada semua pihak yang telah membantu pelaksanaan pengerjaan tugas akhir ini khususnya kepada:

1. Ibu Devi Fitriyah, S.Kom.,MTI selaku dosen pembimbing dan koordinator tugas akhir Jurusan Teknik Informatika Universitas Mercu Buana yang telah memberikan bimbingan, perhatian, pengarahan, dan kesabaran dalam penyusunan tugas akhir ini
2. Bapak Abdusy Syarif, ST., MT, Ahmad Kodar Drs., MT, Nixon Erzed, Ir., MT, serta Ibu Sarwati Rahayu, ST., MMSI selaku dosen penguji
3. Ayahanda S.M Lumbantobing dan Ibunda M br Simanjuntak, kakak-kakak serta adik tercinta yang senantiasa mendoakan, memberi dukungan, dorongan, dan semangat
4. Sahabat ku, Rinto Harianja, yang telah sangat membantu dengan bahan-bahan, diskusi, serta saran di sela-sela kesibukannya
5. Renold Siregar, yang senantiasa mendukung, memberi semangat, dan menemani selama pengerjaan tugas akhir ini

6. Bapak Bambang Tutuka dan Mohammad Satibi, Departemen Teknologi Informasi PT Mane Indonesia, yang memberikan kesempatan untuk menggunakan data operasioanal *Inventory Control* dan *Sales* sebagai kasus uji dalam tugas akhir ini
7. Segenap staf pengajar dan TU Universitas Mercu Buana
8. Pihak-pihak yang tidak dapat disebutkan penulis satu per satu, yang turut membantu dalam proses penyelesaian tugas akhir ini

Semoga tugas akhir ini dapat bermanfaat bagi siapa saja yang berminat dalam bidang basis data khususnya sinkronisasi basis data operasional menjadi *independent data mart*. Penulis menyadari tugas akhir ini masih memiliki kekurangan, untuk itu kritik dan saran yang membangun sangat diharapkan.

Jakarta, 27 Juni 2009

Penulis

## ABSTRACT

Information is the essential part of business development. Instead of to support business flow's efficiency, information is also used for decision support in a company. PT Mane Indonesia is one of company in flavours and fragrances industries that already have an information system to support its operational activities. This system is supported by several operational databases that are not integrated yet. This disintegration causes the problem in providing the information report.

The purpose of this study is to make a synchronisation from several operational databases in PT Mane Indonesia into an independent data mart. Independent data mart it self is a simple form of data warehouse that focused on single subject (functional area) that are built and controlled by a single department within a company.

Synchronization of data, independent data mart's design and implementation are limited to Inventory control dan Sales part in PT Mane Indonesia. At the last stage of this project, independent data mart's process and results are evaluated. The stages that have been done and that are going to be developed will be covered in the last part of this thesis.

*Keywords: Independent data mart, operational databases synchronization*

viii+120 pages; 16 figures; 47 tables; References: 10 (1999-2008)

## ABSTRAK

Informasi merupakan substansi yang sangat penting dalam perkembangan bisnis saat ini. Di samping untuk mendukung efisiensi aliran bisnis, informasi juga digunakan untuk mendukung pengambilan keputusan pada sebuah perusahaan. PT Mane Indonesia merupakan sebuah perusahaan yang bergerak di bidang industri *flavour* dan *fragrance*, telah memiliki sistem informasi untuk mendukung kegiatan operasionalnya. Sistem ini didukung oleh beberapa basis data operasional yang belum terintegrasi, sehingga mengalami masalah untuk menyajikan informasi dalam bentuk laporan.

Tugas akhir ini bertujuan untuk melakukan sinkronisasi data dari beberapa basis data operasional PT Mane Indonesia menjadi *independent data mart*. *Independent data mart* merupakan bentuk sederhana dari *data warehouse* yang fokus terhadap satu bagian ( bagian fungsional) dan dapat dikembangkan dan dikendalikan oleh sebuah bagian pada perusahaan.

Sinkronisasi data, desain, serta *implementasi independent data mart* ini dibatasi pada bagian *Inventory Control* dan *Sales* PT Mane Indonesia. Pada tahap akhir pengembangan *independent data mart* dilakukan pengujian terhadap proses dan hasil *independent data mart*. Tahapan-tahapan yang dilakukan dan yang perlu ditingkatkan diulas pada bagian akhir skripsi ini.

*Kata kunci: Independent data mart, Sinkronisasi basis data operasional*

viii+120 halaman; 16 gambar; 47 tabel; Daftar acuan: 10 (1999-2008)

## DAFTAR ISI

	Halaman
LEMBAR PERNYATAAN.....	2
LEMBAR PERSETUJUAN.....	3
KATA PENGANTAR .....	i
ABSTRACT .....	iii
ABSTRAK.....	iv
DAFTAR ISI .....	v
DAFTAR GAMBAR .....	vii
DAFTAR TABEL.....	viii
BAB I PENDAHULUAN .....	1
1.1 LATAR BELAKANG.....	1
1.2 RUMUSAN MASALAH .....	2
1.3 TUJUAN.....	2
1.4 BATASAN MASALAH .....	3
1.5 METODOLOGI.....	3
1.6 SISTEMATIKA PENULISAN.....	4
BAB IIDASAR TEORI .....	5
2.1 SEJARAH PERKEMBANGAN BASIS DATA .....	5
2.2 KONSEP DAN ISTILAH DALAM DATA WAREHOUSE/ DATA MART .	7
2.2.1 Multidimensional Database.....	7
2.2.2 Hyper Cube .....	8
2.2.3 Struktur Multidimensional Relasional.....	9
2.2.4 Data Acquisition .....	12
2.2.5 Metadata.....	12
2.3 KONSEP <i>INDEPENDENT DATA MART</i> .....	14
2.3.1 Pengertian dan Karakteristik <i>Independent Data Mart</i> .....	14
2.3.2 Langkah-langkah Implementasi <i>Independent Data Mart</i> .....	15
2.4 TEORI KHUSUS.....	18
Analisis SWOT.....	18
BAB IIIANALISIS SISTEM DAN BASIS DATA AS-IS	20
3.1 DESKRIPSI UMUM PT MANE INDONESIA .....	20
3.1.1 Visi dan Misi PT MANE Indonesia .....	21
3.1.2 Struktur Organisasi PT MANE Indonesia .....	21
3.2 ANALISIS SISTEM AS-IS .....	33
3.2.1 Proses Bisnis <i>As-Is</i> .....	33
3.2.2 Teknologi Informasi di PT Mane Indonesia.....	34
3.3 ANALISIS SWOT .....	36
3.4 DESKRIPSI BASIS DATA AS-IS.....	41
3.4.1 ERD ( <i>Entity Relationship Diagram</i> ).....	41



3.4.2	Deskripsi Tabel.....	42
3.5	ANALISIS KEBUTUHAN DATA DAN INFORMASI DI PT MANE INDONESIA .....	57
3.6	MASALAH YANG DIHADAPI.....	58
3.7	ALTERNATIF PEMECAHAN MASALAH .....	60
BAB IVSINKRONISASI BASIS DATA OPERASIONAL MENJADI INDEPENDENT DATA MART.....		62
4.2	ARSITEKTUR INDEPENDENT DATA MART .....	62
4.3	PERANCANGAN INDEPENDENT DATA MART.....	65
4.3.1	Permasalahan dan Kebutuhan Informasi .....	65
4.3.2	Kebutuhan Data Bagian <i>Inventory Control</i> dan <i>Sales</i> .....	66
4.3.3	Klasifikasi Data Untuk Skema Independent Data Mart.....	67
4.3.4	Metadata.....	80
4.4	Proses ETL ( <i>Extraction, Transformation, Loading</i> ).....	90
4.4.1	Extract .....	90
4.4.2	Transformasi .....	97
4.4.3	Loading .....	102
4.5	SISTEM PENDUKUNG.....	106
BAB VANALISIS IMPLEMENTASI INDEPENDENT DATA MART		107
5.1	PENGUJIAN .....	107
5.2	ANALISIS PENGUJIAN.....	118
BAB VI KESIMPULAN DAN SARAN .....		119
6.1	Kesimpulan.....	11319
6.2	Saran .....	120
DAFTAR REFERENSI.....		11621

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Representasi Hypercube.....	9
Gambar 2.2 Contoh Star Schema .....	11
Gambar 2.3 Contoh Snowflake Schema .....	12
Gambar 3.1 Struktur Organisasi PT Mane Indonesia .....	23
Gambar 3.2 Sistem As-Is .....	34
Gambar 3.3 ERD Inventory Control & Sales.....	45
Gambar 4.1 Arsitektur Independent Data Mart .....	63
Gambar 4.2 Skema Basis Data Relasional Inventory Control & Sales .....	68
Gambar 4.3 Skema Bintang Gabungan .....	75
Gambar 4.4 Skema Bintang Stock.....	76
Gambar 4.5 Skema Bintang Internal_Consumption.....	76
Gambar 4.6 Skema Bintang Shipment_Posting.....	77
Gambar 4.7 Skema Bintang Shipment_Posting.....	78
Gambar 4.8 Skema Bintang Sales_Activity.....	79
Gambar 4.9 Skema Bintang Sales_Activity.....	80
Gambar 4.10 Hasil Transformasi.....	102

## DAFTAR TABEL

	Halaman
Tabel 2.1 Matriks SWOT.....	19
Tabel 3.2 Stock In Out.....	46
Tabel 3.3 MontlyPosting_CS.....	47
Tabel 3.4 Product Stock.....	47
Tabel 3.4 Lanjutan-Product Stock.....	48
Tabel 3.5 DO.....	49
Tabel 3.6 DO_Posted.....	50
Tabel 3.7 DO_Collie_Contents.....	50
Tabel 3.7 Lanjutan-DO_Collie_Contents.....	51
Tabel 3.8 Lot_Transaction.....	51
Tabel 3.8 Lanjutan-Lot_Transaction.....	52
Tabel 3.9 Monthly_Rekap.....	52
Tabel 3.10 Invoice_IC.....	53
Tabel 3.11 Invoice_Detail_IC.....	53
Tabel 3.11 Lanjutan-Invoice_Detail_IC.....	54
Tabel 3.12 Sales Invoice Header.....	54
Tabel 3.13 Sales Invoice Detail.....	55
Tabel 3.14 External_List.....	56
Tabel 3.15 Product_Identification.....	57
Tabel 3.16 Category.....	58
Tabel 4.1 Dimensi.....	72
Tabel 4.1 Lanjutan-Dimensi.....	73
Tabel 4.2 Fakta.....	74
Tabel 4.3 Stock_Fact.....	81
Tabel 4.4 Internal_Consumption_Fact.....	82
Tabel 4.5 Shipment_Post_Fact.....	82
Tabel 4.6 Summarize_Transaction_Fact.....	83
Tabel 4.8 Sales_Statistic_Fact.....	83
Tabel 4.8 Lanjutan-Sales_Statistic_Fact.....	84
Tabel 4.9 Agent.....	84
Tabel 4.10 Agent.....	84
Tabel 4.11 Category.....	85
Tabel 4.12 Category_Jual.....	85
Tabel 4.13 Country.....	86

	Halaman
Tabel 2.1 Matriks SWOT.....	19
Tabel 4.14 Currency.....	86
Tabel 4.15 Customer.....	86
Tabel 4.15 Lanjutan-Customer.....	87
Tabel 4.16 Division.....	87
Tabel 4.17 Invoice.....	87
Tabel 4.17 Lanjutan-Invoice.....	88
Tabel 4.18 Motive.....	88
Tabel 4.19 Lanjutan-Period.....	89
Tabel 4.19 Lanjutan-Period.....	89
Tabel 4.20 Product.....	89
Tabel 4.21 Sales.....	90
Tabel 4.22 State.....	90

# **BAB I**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG**

Informasi merupakan substansi yang sangat penting dalam perkembangan bisnis saat ini. Di samping untuk mendukung efisiensi aliran bisnis, informasi juga digunakan untuk mendukung proses pengambilan keputusan pada sebuah perusahaan.

PT Mane Indonesia adalah sebuah perusahaan yang bergerak dalam industri *flavour & fragrances*. Merupakan kantor cabang perusahaan asing bernama V Mane Fils yang berpusat di negara Perancis. Beberapa departemen dibentuk untuk mendukung kegiatan operasional perusahaan dan memiliki fungsi masing-masing. Salah satu diantaranya adalah Departemen Teknologi Informasi.

Saat ini, PT Mane Indonesia memiliki sebuah sistem informasi untuk mendukung kegiatan operasionalnya. Sistem informasi ini didukung oleh beberapa basis data operasional yang belum terintegrasi. Akan tetapi sistem ini belum dilengkapi dengan fasilitas yang memungkinkan pengguna pada level tertentu untuk menampilkan informasi dalam bentuk laporan yang diperlukan untuk mendukung proses pengambilan keputusan.

Perkembangan bisnis yang semakin meningkat mengakibatkan transaksi data juga semakin meningkat. Perkembangan ini mengantar ke sebuah kebutuhan akan sebuah solusi yang dapat menjawab kebutuhan akan penyediaan informasi yang efisien dan terintegrasi.

## 1.2 RUMUSAN MASALAH

Dalam latar belakang telah dijelaskan bahwa pada sistem *as-is* terdapat beberapa masalah yang mempengaruhi proses penyajian informasi yang dibutuhkan dalam proses pengambilan keputusan pada level tertentu di perusahaan. Dari permasalahan yang muncul pada sistem *as-is* di PT Mane Indonesia terdapat rumusan masalah untuk dikaji lebih lanjut yaitu:

1. Basis data operasional yang terletak pada beberapa lokasi yang berbeda. Untuk itu perlu dilakukan integrasi data.
2. Proses pengambilan data dari beberapa basis data operasional yang relatif tidak efisien. Untuk itu perlu diadakan analisa ulang metode pengambilan data.
3. Pada sistem *as-is* belum terdapat fasilitas untuk menampilkan informasi dalam bentuk laporan

## 1.3 TUJUAN

Tujuan yang ingin dicapai dari kegiatan tugas akhir ini adalah:

1. Membuat sebuah basis data yang mendukung penyediaan informasi secara terintegrasi yaitu *independent data mart*
2. Melakukan sinkronisasi data dari basis data sumber ke *independent data mart* untuk menjamin akurasi informasi
3. Membantu pengguna pada level tertentu untuk mendapatkan informasi yang penyajiannya dapat dimanfaatkan dalam proses pengambilan keputusan.

#### 1.4 BATASAN MASALAH

Batasan masalah pada tugas akhir ini adalah:

1. Sumber data yang digunakan dalam kegiatan ini adalah basis data operasional PT Mane Indonesia yang berhubungan dengan bagian *Inventory Control* dan *Sales*
2. Sinkronisasi data dari basis data operasional menjadi *independent data mart* dibatasi hanya untuk data *Inventory Control* dan *Sales*
3. Pengujian *independent data mart* dilakukan terhadap kasus *Inventory Control* dan *Sales*

#### 1.5 METODOLOGI

Metode yang digunakan dalam melakukan tugas akhir ini adalah sebagai berikut:

1. Analisis terhadap sistem dan basis data as-is *Inventory Control* dan *Sales*
2. Eksplorasi dan analisis untuk mengidentifikasi dan mengembangkan *independent data mart*.
3. Eksplorasi dan analisis dilakukan dengan cara studi literatur dan pencarian bahan di internet.
4. Perancangan skema *independent data mart* dan proses-proses pembangunannya.
5. Eksplorasi perangkat lunak untuk pengujian yang sesuai dengan pembangunan *independent data mart*.

6. Pengujian *independent data mart* dengan menggunakan perangkat lunak yang telah dipilih dalam butir 5.

## 1.6 SISTEMATIKA PENULISAN

- Bab I Berisi latar belakang, rumusan masalah, metodologi, serta tujuan dan batasan yang akan digunakan.
- Bab II Berisi dasar teori yang diperlukan seperti konsep dalam *data warehouse/ data mart*, konsep *independent data mart*, serta teori khusus yang digunakan untuk analisis bisnis
- Bab III Berisi analisis sistem dan basis data *as-is*
- Bab IV Berisi deskripsi sinkronisasi basis data operasional sistem *as-is* menjadi *independent data mart*
- Bab V Berisi deskripsi implementasi pembangunan *independent data mart*, pengujian, serta analisis hasil pengujian *independent data mart*
- Bab VI Berisi kesimpulan yang dapat diambil dari pelaksanaan tugas akhir dan saran yang bisa diberikan untuk studi lebih lanjut dalam bidang *independent data mart*.



## BAB II

### DASAR TEORI

#### 2.1 SEJARAH PERKEMBANGAN BASIS DATA

Basis data merupakan perintis *data ware house* dan *data mart*. Sejarah perkembangannya diawali dengan *file based system* yang diimplementasikan sejak akhir tahun 60an, pada saat teknologi komputer mulai dikenal. Sistem basis data tradisional adalah berupa penggunaan file untuk menyimpan data baik dalam bentuk dua dimensi atau struktur tabular. Program yang akan memproses data ini harus mengetahui relasi aktual antara bagian data yang berbeda-beda sebelumnya. Relasi data harus dikodekan ke dalam program. Hal ini membutuhkan integrasi data yang sangat tinggi dengan program, seperti penggunaan *data division* pada COBOL atau *structure* pada C. Keterbatasan sistem ini adalah kurang skalabilitas, sekali diimplementasikan, untuk menambahkan fitur atau skema data harus dilakukan pengkodean lagi. Dan juga, apabila diperlukan pertukaran data antar program, maka sekali lagi harus dilakukan pengkodean untuk memindahkan data. Sehingga tidak ada standar.

Kemudian muncul *hierarchical database system*, model basis data ini berkembang hampir pada saat yang bersamaan dengan *file based system*, akan tetapi baru banyak digunakan pada tahun 80an. Pola penyimpanan data seperti *binary tree* yang menggambarkan relasi data *one-to-many*. Sistem ini berhasil mengatasi keterbatasan *file based system* terutama di bagian independensi data, akan tetapi sangat sulit untuk

diimplementasikan. Kemudian muncul *network database system*. Sistem ini berkembang beberapa lama setelah kedua sistem sebelumnya, sesuai dengan CODASYL (*Conference on Data System Languages*) tersusunlah *record* dan *set*. *Record* menyimpan data aktual, dan *set* mendefinisikan relasi antar data secara terpisah. Sistem ini memiliki performansi yang baik pada pengambilan data akan tetapi sulit untuk diimplementasikan dan dirubah.

*Relational Database Management System* ( DBMS relasional ) adalah sistem basis data modern yang dipakai hingga saat ini, yang sepertinya sudah dapat mengatasi sebagian besar batasan sistem pendahulunya. Pada prinsip RDBMS, data disimpan dalam bentuk tabel, dan tabel memiliki relasinya. Data disusun dengan cara yang diingankan oleh beberapa perangkat lunak DBMS, dengan menggunakan sintaks SQL. DBMS kemudian merespon dengan menyediakan hasil kepada program yang memanggilnya. DBMS memungkinkan desain dari sistem tertentu untuk menjaga program dan basis data secara terpisah. Hal ini memungkinkan pertukaran data antar program tanpa membutuhkan pemrograman atau perancangan masing masing atau setiap interaksi. Ada banyak penyedia perangkat lunak yang menyediakan relational DBMS seperti SQL Server, MySQL, PostgreSQL, MSSQL, Sybase, Oracle. Dan hampir semua sistem ini dikembangkan berdasarkan fitur dasar RDBMS.

Banyak model baru sedang dikembangkan dengan sistem yang lebih kompleks dan membuat interaksi basis data semakin efisien. *object relational system*, *entity relationship system*, *non-normalized relational system* adalah beberapa kata teknis yang beredar di kalangan ahli basis data saat ini.

## 2.2 KONSEP DAN ISTILAH DALAM DATA WAREHOUSE/ DATA MART

*Data warehouse/ data mart* merupakan salah satu cara untuk menyediakan solusi dalam bidang analisis bisnis. Dalam perkembangannya, terdapat beberapa konsep dan istilah yang berlaku baik untuk *data warehouse* dan *data mart*.

### 2.2.1 Multidimensional Database

*Data warehouse/ data mart* digunakan sebagai alat untuk mengambil keputusan melalui analisis bisnis berdasarkan sumber data operasional perusahaan yang biasanya tersedia dalam bentuk basis data relasional dan sudah ternormalisasi untuk meminimasi redundansi data, resiko hilangnya data dan data eror. Basis data relasional yang ternormalisasi ini memiliki beberapa kekurangan sehingga tidak dapat digunakan secara langsung untuk analisis. Beberapa kekurangan tersebut adalah:

1. Sulit dimengerti oleh pengguna

Struktur data yang telah dinormalisasi tidak dapat memetakan secara sederhana proses pemikiran pengguna. Di lain pihak pengguna *data warehouse/ data mart/ OLAP* adalah pengguna awam yang berasal dari kalangan bisnis, sehingga struktur ini tidak *friendly* dan tidak sederhana bagi pengguna.

2. Memerlukan pengetahuan SQL

3. Tidak teroptimasi untuk mensupport query untuk pengambilan keputusan

4. Basis data operasional tidak memelihara history data

Terdapat dua pendekatan dalam membangun struktur multidimensional database ini yaitu:

1. Struktur multidimensional khusus, contohnya *hypercube*

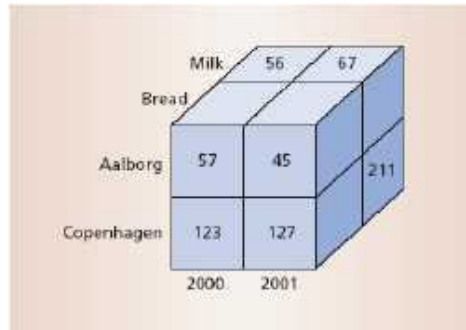
2. Struktur multidimensional relasional, contohnya *star schema*, *snowflake schema*, dan sebagainya.

### 2.2.2 Hyper Cube

Sebuah *hypercube* adalah sekelompok data *cell* yang disusun oleh dimensi-dimensi data. Sedangkan sebuah dimensi didefinisikan sebagai suatu atribut terstruktur dari sebuah *hypercube* yang berupa sebuah *list of member* yang semuanya memiliki tipe yang sama dalam sudut pandang pengguna terhadap data. Dimensi digunakan untuk dua tujuan, yaitu seleksi data dan pengelompokan data pada level yang detail yang diinginkan. *Hypercube* secara fisik disimpan dalam bentuk array multidimensional. Arsitektur dasar penyimpanan data pada OLAP yang didasarkan pada bentuk *hypercube* ini biasa disebut sebagai Multidimensional OLAP (MOLAP). Sampai saat ini belum ada standar dalam pembangunan *hypercube*, sehingga cara pembangunan *hypercube* merupakan rahasia perusahaan.

Pada implementasi *hypercube* memiliki fleksibilitas yang rendah karena bila terjadi penambahan dimensi maka perlu dilakukan pembangunan ulang. Namun *hypercube* memiliki kecepatan pemrosesan yang sangat baik untuk kategori basis data dengan ukuran kecil hingga menengah. Representasi *hypercube* dapat dilihat pada gambar 2.1

Representasi Hypercube



Gambar 2.1 Representasi Hypercube

### 2.2.3 Struktur Multidimensional Relasional

Struktur multidimensional relasional hadir sebagai jawaban atas perlunya basis data untuk analisis yang dikembangkan dari struktur relasional yang telah ada. Struktur dimensional relasional dapat dibagi menjadi beberapa skema antar lain adalah *star schema* dan *snowflake schema*. Arsitektur dasar penyimpanan data pada OLAP yang didasarkan pada bentuk relasional ini biasa disebut sebagai *Relational OLAP (ROLAP)*.

#### 2.2.3.1 Star Schema

*Star schema* biasa disebut juga sebagai model dimensional dan dipopulerkan oleh *Ralph Kimball*.

Terdapat dua tabel yang mendasari model dimensional ini yaitu:

1. Tabel Fakta (*Fact Table*)

*Fact table* merupakan *primary* tabel yang digunakan untuk menyimpan fakta aktual atau ukuran performansi dalam bisnis. Fakta adalah data *item* yang numerik yang merepresentasikan kepentingan bisnis, digunakan oleh *user* untuk

menganalisis dan menarik kesimpulan agar mendapatkan pengertian yang lebih baik akan bisnis yang ada.

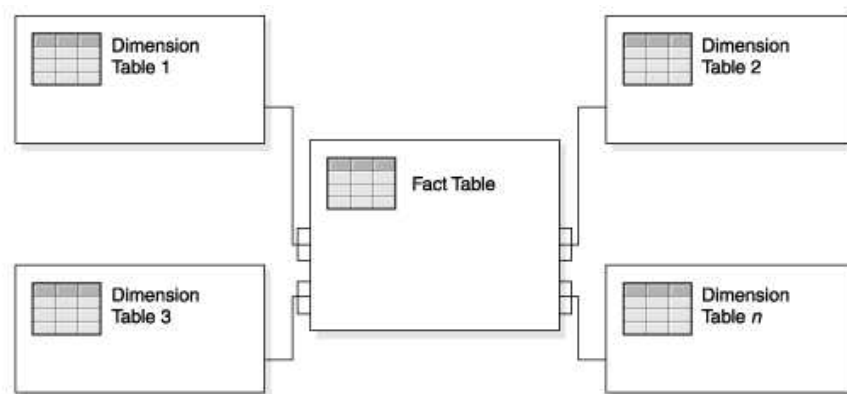
Beberapa contoh fakta dalam beberapa jenis bisnis:

- Retail: jumlah unit yang terjual, jumlah penjualan
- Telekomunikasi: lama percakapan dalam menit, rata-rata lama percakapan
- Bank: jumlah transaksi
- Penerbangan: harga tiket, berat bagasi

## 2. Tabel Dimensi

Tabel dimensi merupakan tabel yang menyusun konteks dari fakta, dengan kata lain tabel dimensi menyimpan keterangan yang mendeskripsikan fakta dari bisnis (metadata).

Dalam *star schema* sebuah dimensi direpresentasikan dengan sebuah tabel dimensi yang berisi atribut.

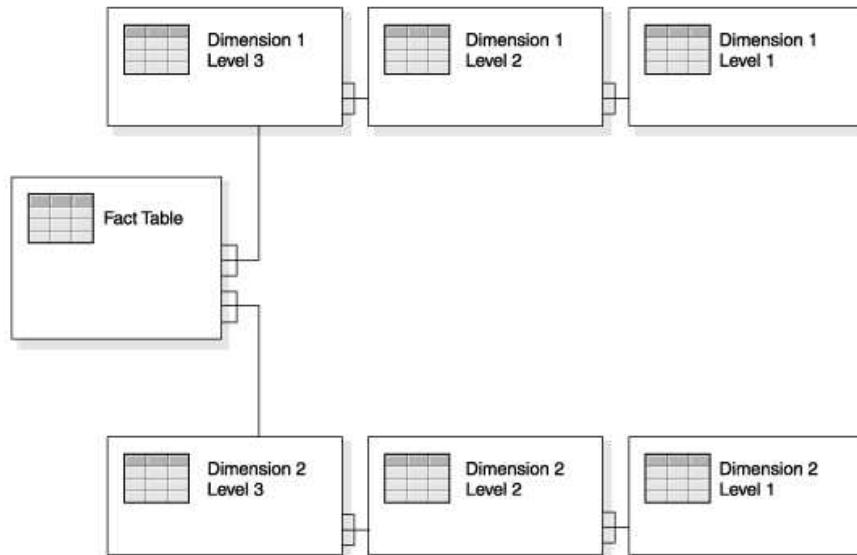


**Gambar 2.2 Contoh Star Schema**

### 2.2.3.2 Snowflake Schema

*Snowflake schema* merupakan varian dari *star schema* yang beberapa tabel dimensinya dipecah/ dinormalisasi kembali menjadi beberapa tabel. Penggambaran graf yang dihasilkan skema ini menyerupai kepingan salju (*snow flake*).

Normalisasi beberapa tabel dalam *snowflake schema* dilakukan untuk mengurangi redundansi data dalam data *warehouse/ data mart* sehingga dapat menghemat dan mempermudah penyimpanan dalam *storage space*. Namun normalisasi ini mengakibatkan semakin kompleksnya proses *join* yang harus dilakukan terhadap data dan semakin rumitnya gambaran presentasi yang didapatkan pengguna. Pada kenyataannya masalah penghematan *storage space* masih dapat ditangani sehingga kecenderungan penerapan *snow flake schema* pada dunia nyata jarang ditemukan karena pengguna lebih memilih untuk mempercepat proses *browsing query* dan menyederhanakan struktur *query*.



Gambar 2.3 Contoh Snowflake Schema

#### 2.2.4 Data Acquisition

*Data acquisition* merupakan salah satu tahap dalam pembangunan *data warehouse/ data mart*.

Beberapa tahapan dalam *data acquisition* adalah:

- Identifikasi sumber data yang relevan terhadap subjek area yang dibangun
- Pembangunan strategi ekstraksi

Dalam tahap ini dilakukan proses penyeleksian data-data dan informasi yang relevan bagi keperluan analisis dan pengambilan keputusan dari beberapa sumber data dan *platform* yang berbeda

- Pentransformasian sumber data pada target spesifik

Dalam tahap ini dilakukan transformasi dan integrasi skema serta struktur yang berbeda-beda ke dalam skema yang terstruktur yang terdefenisi dalam *data warehouse*.

- *Loading* sumber data ke dalam target yang telah didefenisikan

Dilakukan proses pemindahan data secara fisik dari sistem operasional ke dalam *data warehouse*.

Tahap *data acquisition* ini juga sering disebut sebagai proses ETL (Ekstraksi, Transformasi, *Loading*).

#### 2.2.5 Metadata

Metadata merupakan unsur yang sangat penting untuk menjaga integritas data *warehouse/data mart*, karena metadata menyimpan informasi kejadian yang terjadi



pada proses bisnis. Informasi ini dapat digunakan untuk bisnis maupun *technical user*.

Beberapa pengertian metadata adalah:

1. Data yang menjelaskan tentang data
2. Seluruh informasi dalam lingkungan *data warehouse/ data mart* yang bukan merupakan data aktual itu sendiri.
3. Deskripsi dari struktur, *content, keys*, indeks dll dari data
4. Seluruh data fisik ( terdapat dalam *software* dan media lain) dan pengetahuan (berasal dari pekerja dan berbagai media) dari dalam dan luar organisasi, termasuk informasi mengenai data fisik, teknik dan proses bisnis, aturan dan batasan tentang data dan struktur dari data yang digunakan oleh perusahaan.

Metadata dalam *data warehouse/ data mart* dapat terdiri atas:

- a. Metadata Administratif
  1. Deskripsi basis data sumber (skema dan content)
  2. Deskripsi *gateway*
  3. Global skema/ skema data *warehouse (view, definisi, deskripsi data)*
  4. Pendefinisian tabel dimensi, hirarki, dan fakta
  5. *Pre-defined query* dan *report*
  6. Pendefinisian data mart (jika ada)
  7. Pemartisian data
  8. Aturan ekstraksi, pembersihan, transformasi data dan pemberian *default value*.
  9. Aturan *refresh*.
  10. Karakteristik pengguna, *profile*, dan *security*.

b. Metadata operasional

1. Data *lineage* (*history* dan kronologis)
2. Kekinian data (aktif, *archieved*, *purged*)
3. Informasi monitoring (statistik penggunaan/proses, laporan kesalahan)

c. Metadata Bisnis

1. Terminologi dan definisi data dari sudut pandang bisnis
2. Daftar pemilik data
3. *Charging policies*

## 2.3 KONSEP *INDEPENDENT DATA MART*

### 2.3.1 Pengertian dan Karakteristik *Independent Data Mart*

*Independent data mart*, disebut juga sebagai *stovepipe data mart*, merupakan suatu arsitektur *data mart* yang berkembang dan masih sering dipakai sebagai alat bantu pengambilan keputusan dalam dunia bisnis.

Berdasarkan beberapa literatur, pengertian *independent data mart* adalah:

- *Independent data mart*: *data mart* yang dibangun secara terpisah dari *data warehouse* pusat, tidak dibangun dari *data warehouse* maupun tidak menjadi sumber data untuk sebuah *data warehouse*. *Independent data mart* dirancang dan dibangun terutama sebagai solusi takstis atas DSS sebuah departemen.
- *Stovepipe data mart*: biasa disebut sebagai *independent data mart*. *Stovepipe data mart* tidak terintegrasi dengan metadata pusat (sentral). Tidak memiliki aturan bisnis, semantik dan definisi yang sama (*common*). *Stovepipe data mart* mendukung kebutuhan individual unit bisnis, tetapi tidak mendukung

kebutuhan individual unit bisnis, tetapi tidak mendukung strategi *enterprise* dan data *warehouse*.

*Independent data mart* memiliki beberapa sifat sebagai berikut:

1. Setiap data mart bersumber secara langsung dari sistem operasional tanpa menggunakan suatu struktur data *warehouse*.
2. Masing-masing data mart untuk setiap departemen dalam suatu perusahaan yang sama dibangun oleh tim secara independen dan terpisah, dan masing-masing tim tersebut menggunakan *tools*, perangkat lunak, perangkat keras, dan proses yang berbeda-beda.

### **2.3.2 Langkah-langkah Implementasi Independent Data Mart**

Tahapan-tahapan utama dalam implementasi *independent data mart* adalah merancang skema, membangun *physical storage*, mengisi *independent data mart* dengan data yang berasal dari *source system*, melakukan akses terhadap *independent data mart* untuk mengambil sebuah keputusan yang bermanfaat dan melakukan pengelolaan yang baik dari waktu ke waktu. Tahapan-tahapan pembangunan *independent data mart* dapat dilihat lebih jelas seperti uraian dibawah ini:

#### *1. Designing*

*Design* merupakan langkah awal dalam proses *independent data mart*. Tahapan ini mencakup tahapan inialisasi permintaan akan pengembangan *independent data mart* dengan cara pengumpulan informasi tentang kebutuhan data, serta pengembangan disain logis dan fisik.

Tahapan disain melibatkan tugas-tugas berikut ini:

- a. Mengumpulkan kebutuhan bisnis dan kebutuhan teknis
- b. Identifikasi sumber data
- c. Seleksi bagian data yang penting
- d. Merancang struktur logis dan fisik *independent data mart*

## 2. *Constructing*

Tahapan ini meliputi pembuatan basis data fisik dan struktur logis dihubungkan dengan independent data mart untuk menyediakan akses data yang cepat dan efisien.

Tahapan ini terdiri langkah-langkah berikut ini:

- Membuat basis data fisik dan struktur penyimpanan data seperti tempat penyimpanan tabel, dihubungkan dengan independent data mart
- Membuat objek skema, seperti tabel-tabel dan indeks yang telah didefinisikan pada tahap desain
- Memutuskan cara terbaik untuk menyiapkan tabel dan mengakses struktur data

## 3. *Populating*

Tahapan ini mencakup semua bagian yang berhubungan dengan pengambilan data dari sumber, memilih data yang tepat, memodifikasi format data ke dalam bentuk yang tepat, dan memindahkan data ke dalam *independent data mart*. Secara formal, tahapan populating meliputi langkah-langkah berikut ini:

- Menentukan sumber data dengan struktur data target
- Melakukan ekstraksi data
- Memilih dan melakukan transformasi data

- Memuat data ke dalam *independent data mart*
- Membuat dan menyimpan metadata

#### 4. *Accessing*

Tahapan ini mencakup menyediakan data yang akan digunakan, meng-*query* kan data, menganalisa, membuat *report*, membuat grafik, dan menampilkannya. Biasanya *end user* menggunakan tampilan grafis untuk mengajukan *query* ke basis data dan menampilkan hasil *query* tersebut. Langkah-langkah yang diperlukan pada tahapan ini adalah sebagai berikut:

- Membuat *intermediate layer* sebagai alat *front-end*. Layer ini menerjemahkan struktur basis data dan nama objek ke *business term*, sehingga *end user* dapat berinteraksi dengan *independent data mart* dengan menggunakan istilah-istilah yang berhubungan dengan *business function*.
- Memelihara dan mengatur *business interface*
- Membangun dan mengatur struktur basis data, seperti meringkaskan tabel yang akan membantu proses eksekusi *query* yang dimasukkan melalui interface lebih cepat dan efisien.

#### 5. *Managing*

Tahapan ini mencakup pemeliharaan *independent data mart* selama *independent data mart* tersebut masih ada. Langkah-langkah yang perlu dilakukan pada tahapan ini adalah sebagai berikut:

- Menyediakan cara akses data yang aman
- Mengatur pertumbuhan data

- Mengoptimalkan sistem untuk performansi yang lebih baik
- Memastikan ketersediaan data meskipun sistem mengalami kegagalan

## 2.4 TEORI KHUSUS

### Analisis SWOT

Analisis SWOT adalah identifikasi berbagai faktor secara sistematis untuk merumuskan strategi, berdasarkan logika yang dapat memaksimalkan kekuatan (Strengths) dan peluang (Opportunities), dan secara bersamaan dapat meminimalkan kelemahan (Weaknesses) dan ancaman (Threats). Jadi, analisis SWOT membandingkan antara faktor eksternal; peluang dan ancaman dengan faktor internal; kekuatan dan kelemahan. Matriks SWOT menampilkan delapan kotak, yaitu dua kotak sebelah kiri menampilkan faktor eksternal (peluang dan ancaman), dua kotak paling atas menampilkan faktor internal (kekuatan dan kelemahan) dan empat kotak lainnya merupakan isu-isu strategis yang timbul sebagai hasil pertemuan antara faktor eksternal dan internal.

Menurut Setiawan Hari Purnomo dan Zulkieflimansyah (1999), berdasarkan hasil analisis SWOT, terdapat empat alternative strategi yang tersedia yaitu strategi SO, WO, ST, dan WT. Matriks SWOT digambarkan sebagai berikut:

	<b>Opportunity</b>	<b>Threat</b>
<b>Strenght</b>	SO Strategies	ST Strategies
<b>Weakness</b>	WO Strategies	WT Strategies

**Tabel 2.1 Matriks SWOT**

Keterangan Matriks SWOT tersebut sebagai berikut :

**SO strategies:** merupakan situasi yang menguntungkan. Perusahaan memiliki peluang dan kekuatan sehingga dapat memanfaatkan peluang yang ada. Strategi yang harus diterapkan dalam kondisi ini adalah mendukung kebijakan pertumbuhan yang agresif (*growth oriented strategy*).

**ST strategies :** dalam situasi ini perusahaan menghadapi berbagai ancaman, tetapi masih memiliki kekuatan dari segi internal. Strategi yang harus diterapkan dalam kondisi ini adalah menggunakan kekuatan untuk memanfaatkan peluang jangka panjang dengan cara strategi diversifikasi (produk/pasar).

**WO strategies:** dalam situasi ini perusahaan menghadapi peluang pasar yang besar, tetapi juga menghadapi beberapa kendala/ kelemahan internal. Fokus strategi pada situasi ini adalah meminimalkan masalah-masalah internal sehingga dapat merebut peluang yang lebih baik.

**WT strategies:** merupakan situasi yang tidak menguntungkan, sehingga perusahaan harus menghadapi berbagai ancaman dan kelemahan internal.

## BAB III

### ANALISIS SISTEM DAN BASIS DATA AS-IS

#### 3.1 DESKRIPSI UMUM PT MANE INDONESIA

PT Mane Indonesia merupakan cabang perusahaan V.M Fils (Victor Mane & Fils) yang bertempat di Bar sur Loup, Perancis. V.M Fils (Victor Mane & Fils) sudah menjual produk nya lebih dari 25 tahun di Indonesia. Pemasarannya mencakup bahan-bahan pembuatan kosmetik, sabun, deterjen, makanan, dan juga kterek.

Demi meningkatkan pelayanan kepada pelanggan, pada tahun 1995 V.M Fils (Victor Mane & Fils) memutuskan untuk mengembangkan bisnis di Asia Tenggara. Indonesia merupakan negara yang sangat potensial untuk pemasaran sehingga dipilih menjadi lokasi kantor *head quarter* V.M Fils (Victor Mane & Fils) di Asia Tenggara.

PT Mane Indonesia didirikan di awal tahun 1996 dan mulai aktif menjalankan proses produksi pada pertengahan tahun 1998. Terdapat sekitar 200 tenaga kerja di PT Mane Indonesia yang bekerja di bagian *sales, research and development*, produksi, dan administrasi.

Sebagian besar produk yang dihasilkan dijual kepada pelanggan yang berada di Indonesia akan tetapi ada juga yang di ekspor ke Thailand, China, Filipina, Singapura, dan Malaysia.

Pabrik PT Mane Indonesia dilengkapi dengan peralatan produksi yang mampu menghasilkan 700 ton *flavour* dan 200 ton *fragrance*. Hal ini dimungkinkan dengan



pengembangan bangunan pabrik 3 kali lipat dari kondisi bangunan awal didirikan, pengembangan bangunan kantor dan laboratorium 2 kali lipat dari bangunan awal.

Dalam rangka memperkuat citra dan meningkatkan efisiensi, PT Mane Indonesia telah mengimplementasikan *quality management system* sesuai dengan standar ISO 9000. Dan kini, sertifikasi ISO 9001:2000 sebagai wujud komitmen PT Mane Indonesia terhadap kualitas pelayanan dan peningkatan kerjasama dengan pelanggan dan rekanan bisnis.

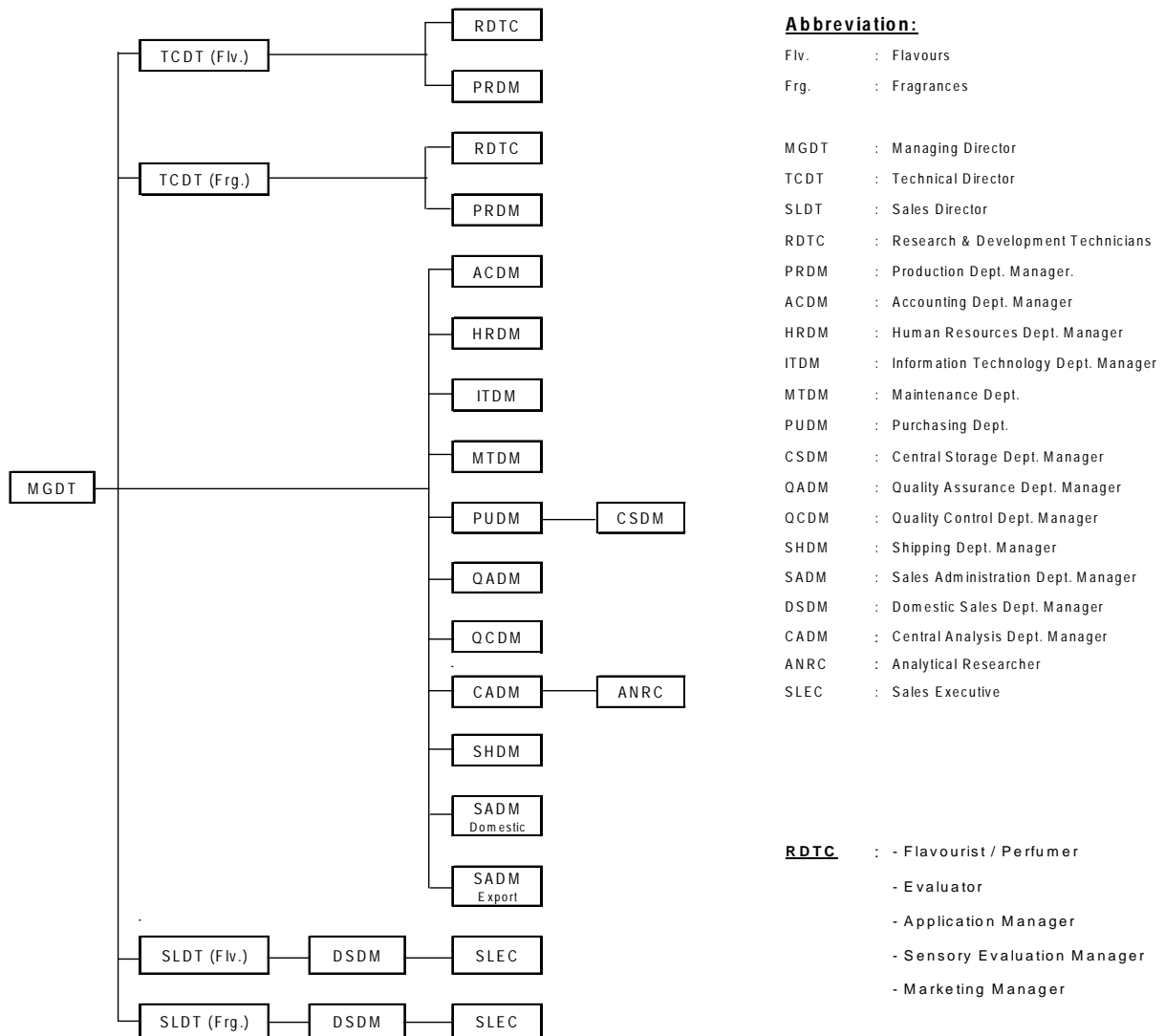
### **3.1.1 Visi dan Misi PT MANE Indonesia**

Sebagai perusahaan yang bergerak di bidang industri *flavour* dan *fragrance*, PT Mane Indonesia memiliki visi dan misi sebagai berikut:

- Memberikan pelayanan terbaik kepada pelanggan sesuai dengan atau bahkan melebihi ekspektasi pelanggan
- Menjadi yang terbaik di industri *flavour* dan *fragrance* dengan menciptakan hal-hal baru, menghasilkan kesuksesan baru, dan menghasilkan produk yang lebih inovatif dari *competitor*
- Mencapai standar kualitas kelas dunia dengan cara mengimplementasikan *Quality and Safety Management System*

### **3.1.2 Struktur Organisasi PT MANE Indonesia**

PT Mane Indonesia memiliki beberapa divisi dalam mendukung kegiatan operasionalnya. Struktur organisasi PT Mane Indonesia digambarkan pada gambar 3.1 Struktur organisasi PT Mane Indonesia.



**Gambar 3.1 Struktur Organisasi PT Mane Indonesia**

Berikut ini merupakan penjelasan tugas dan tanggung jawab masing-masing bagian yang tercakup pada struktur organisasi PT Mane Indonesia.

## 1. MGDT (*Managing Director*)

Managing Director merupakan jabatan tertinggi di PT Mane Indonesia yang memiliki tanggung jawab sebagai berikut:

- Meningkatkan kebijakan dan sasaran mutu PT Mane Indonesia
- Menetapkan dan menyesuaikan struktur organisasi agar mengikuti dan mencapai kebijakan dan sasaran mutu
- Memelihara *Quality Management System* yang disesuaikan dengan permbangan perusahaan dan meningkatkan efektifitas nya secara berkelanjutan
- Mengidentifikasi dan menyediakan sumber daya yang dibutuhkan dalam penerapan dan pemeliharaan *Quality Management System*
- Menyampaikan pentingnya hubungan baik dengan pelanggan kepada pihak perusahaan
- Menunjuk perwakilan manajemen dan memberinya tanggung jawab untuk melakukan komunikasi dengan pihak luar menyangkut segala urusan yang berhubungan dengan mutu sistem PT Mane Indonesia
- Memberi persetujuan terhadap pedoman mutu, prosedur, dan perubahannya
- Memastikan pengetahuan yang terbaik, aplikasi, dan pemeliharaan mutu sistem pada semua level yang ada di PT Mane Indonesia
- Memastikan proses komunikasi yang sesuai untuk diterapkan di perusahaan dan komunikasi tersebut sesuai dengan efektifitas *Quality Management System*

## 2. TCDT (*Technical Director*)

- Memastikan perencanaan dan pengendalian perancangan dan pembuatan produk sesuai dengan harapan pelanggan
- Mengawasi semua hal yang bersifat teknis yang berhubungan dengan penghasilan dan pembuatan produk
- Menjadi perantara pimpinan bagian produksi dengan bagian *Research & Development* dengan pimpinan perusahaan
- Ikut serta dalam penyelesaian masalah, apabila dibutuhkan

## 3. SLDT (*Sales Director*)

- Mengawasi semua hal yang menyangkut penjualan dalam maupun luar negeri
- Memastikan komunikasi yang efektif dengan pelanggan
- Memantau informasi yang berhubungan dengan kepuasan pelanggan
- Menjadi perantara pimpinan *sales* dalam dan luar negeri dengan pimpinan perusahaan
- Ikut serta dalam penyelesaian masalah, apabila dibutuhkan

## 4. RDTC (*Research and Development Technician*)

- Mengawasi tahap perancangan dan pembuatan serta mengadakan perubahan, verifikasi dan validasi perancangan dan pembuatan produk
- Membuat kode bahan baku, barang setengah jadi, dan barang jadi
- Menetapkan kondisi penyimpanan untuk bahan baku, barang setengah jadi, dan barang jadi

- Mengadakan evaluasi terhadap bahan baku dari pemasok dan memastikan kualitas nya
- Menentukan standar bahan baku, barang setengah jadi, dan barang jadi
- Membuat urutan pencampuran bahan baku yang diperlukan dalam membuat barang setengah jadi dengan/atau barang jadi
- Ikut serta dalam penyelesaian masalah yang berhubungan dengan produk, apabila diperlukan

5. PRDM (*Production Dept. Manager*)

- Membuat prosedur dan instruksi kerja yang sesuai dengan *Quality Management System* dan norma tertentu
- Memastikan pengetahuan yang baik dan pelaksanaan prosedur dan instruksi kerja yang berhubungan dengan produksi adalah sesuai dengan *Quality Management System*
- Memastikan bahwa cara memproses untuk proses produksi *flavour* dan *fragrance* sesuai dengan spesifikasi yang ditetapkan dalam dokumen *Quality Management System*
- Memastikan bahwa semua instruksi kerja yang dibutuhkan dan berhubungan dengan proses produksi tersedia di lokasi pada saat produksi dilaksanakan
- Menyusun dan menyimpan arsip hasil pelaksanaan dokumen *Quality Management System*
- Menyediakan informasi yang diperlukan untuk memudahkan pelaksanaan program pemeliharaan *preventive* tahunan

- Ikut serta dalam penyelesaian masalah yang berhubungan dengan produk, apabila diperlukan

6. HRDM (Human Resources Dept. Manager)

- Membuat prosedur dan instruksi kerja yang diperlukan sesuai dengan *Quality Management System* dan aturan yang ditetapkan
- Memastikan ilmu pengetahuan yang baik dan penerapan akan *Quality Management System* dan prosedur dan instruksi kerja yang berhubungan dengan departemen pengelolaan sumber daya manusia perusahaan
- Memastikan bahwa karyawan PT Mane Indonesia menyadari peranan masing-masing dan bagaimana memberikan kontribusi untuk mencapai tujuan sasaran mutu
- Menyusun dan menyimpan arsip hasil pelaksanaan dokumen *Quality Management System*
- Melaksanakan penerimaan tenaga kerja yang dibutuhkan perusahaan sesuai dengan profil yang diberikan pimpinan perusahaan dan mengikuti dokumen mutu
- Mengkoordinasikan jenis pelatihan yang perlu untuk dilaksanakan untuk karyawan PT Mane Indonesia
- Melakukan evaluasi hasil pelatihan dan melaporkan kepada pihak manajemen perusahaan, sedikitnya sekali dalam setahun

7. ITDM (*Information Technology Dept. Manager*)

- Mengembangkan dan memodifikasi sistem untuk mendukung *Quality Management System*
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen teknologi informasi
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*
- Memastikan kehandalan informasi untuk menjamin kebenaran hasil yang disediakan
- Melakukan pemeliharaan terhadap sistem informasi yang ter-*install* dan memastikan program komputer yang dibutuhkan mencukupi
- Mengadakan dan memelihara perangkat komputer perusahaan

8. MTDM (*Maintenance Dept. Manager*)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen *maintenance*
- Memelihara prasarana yang dibutuhkan dalam menghasilkan produk yang sesuai

- Membuat dan mengkoordinasikan pelaksanaan program pemeliharaan tahunan untuk menjamin hasil terbaik pemakaian dan pemasangan peralatan

9. PUDM (Purchasing Dept. Manager)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen *purchasing*
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*
- Melakukan pemesanan secara lokal maupun import sesuai dengan dokumen yang berhubungan
- Memantau kehandalan pemasok dan mencatat performansinya
- Memastikan dokumen *purchasing* mengandung dan menjelaskan dengan jelas karakteristik barang yang dipesan

10. CSDM (*Central Storage Dept. Manager*)

- Menangani penerimaan barang dan memeriksa penampilannya
- Menyiapkan *retain* yang dibutuhkan untuk *quality control*
- Menangani barang di lokasi penyimpanan dan menyimpannya pada kondisi yang sesuai
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*



- Melakukan pemeriksaan gudang secara fisik sesuai yang telah ditetapkan pada dokumen mutu

#### 11. QADM (Quality Assurance Dept. Manager)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen terkait
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*
- Mengelola dan memantau perkembangan program-program peningkatan
- Mengelola dan memantau informasi yang berhubungan dengan kepuasan pelanggan
- Mengekoordinasi pelaksanaan dan pemeliharaan *Quality Management System*
- Mengawasi distribusi dan penambahan dokumen *Quality Management System*
- Menyiapkan program tahunan berupa pemeriksaan secara internal
- Menyediakan informasi yang diperlukan oleh pimpinan perusahaan untuk merubah *Quality Management System*, dan memastikan efektifitas perubahan tersebut memadai untuk ditetapkan
- Berperan serta dalam tindakan pencegahan dan/atau perbaikan

- Mengkoordinasikan pembelajaran, penerapan, dan pemeliharaan *Safety Management System*

## 12. QCDM (Quality Control Dept. Manager)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen terkait
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*
- Mengadakan analisis dan pemeriksaan yang diperlukan untuk setiap produk sesuai dengan yang ditetapkan pada *quality system procedures*
- Memberi identifikasi pada pemeriksaan dan status pengujian semua produk sesuai dengan yang tertera pada *quality system procedures*
- Mengkoordinasikan penerapan semua pemeriksaan dan pengukuran semua peralatan yang digunakan pada proses produksi, penyimpanan dan *quality control* sesuai dengan yang ditetapkan pada *quality system procedures*
- Menerima *retain* dari semua bahan baku, barang setengah jadi dan barang jadi
- Menyatakan produk yang tidak sesuai dan memberitahukannya ke departemen-departemen terkait
- Ikut serta dalam penyelesaian masalah yang berhubungan dengan produk, apabila diperlukan

### 13. SHDM (Shipping Dept. Manager)

- Menyimpan barang jadi pada kondisi yang sebaiknya sebelum dikirimkan kepada pelanggan
- Menangani pengiriman barang jadi kepada pengirim sesuai kebutuhan departemen administrasi *sales*
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*

### 14. SADM (*Sales Administration Dept. Manager*)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen administrasi *sales*
- Menentukan dan menerapkan cara komunikasi yang efektif dengan pelanggan
- Menyusun dan menyimpan dokumen hasil pelaksanaan dokumen *Quality Management System*
- Meninjau semua kebutuhan permintaan pelanggan dengan cara memeriksanya dengan departemen-departemen terkait
- Memberikan tanda terima pesananan kepada pelanggan
- Mengkoordinasikan secara internal pemrosesan permintaan pelanggan dengan departemen-departemen terkait

- Menangani masalah administrasi yang berhubungan dengan pengiriman barang kepada pelanggan

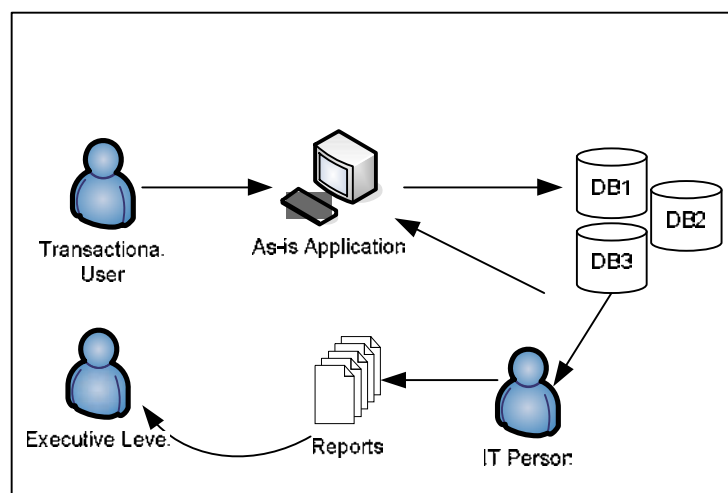
15. SLDT (*Sales Department Manager*)

- Membuat prosedur dan instruksi kerja sesuai dengan yang telah ditetapkan oleh *Quality Management System* dan aturan yang ditentukan
- Melengkapi dan menyimpan dokumen dan berkas yang membutuhkan revisi, perubahan, dan membuat dokumen baru sesuai dengan yang telah ditetapkan pada *Quality System Procedure*
- Menentukan dan menerapkan komunikasi yang efektif dengan pelanggan
- Memastikan penerapan dan pengetahuan yang baik akan prosedur dan instruksi kerja *Quality Management System* yang berhubungan dengan departemen *sales* baik domestik maupun ekspor.

## 3.2 ANALISIS SISTEM AS-IS

### 3.2.1 Proses Bisnis As-Is

Pada sistem yang berjalan sekarang, PT Mane Indonesia telah memiliki sistem informasi untuk mendukung kegiatan operasional perusahaan. Dalam pengoperasiannya, *transactional user* melakukan input dan akses data melalui aplikasi *as-is* yang dikembangkan oleh departemen teknologi informasi perusahaan. Data transaksi disimpan ke dalam beberapa basis data operasional yang terletak di lokasi yang berbeda. Pada sistem yang berjalan sekarang, belum terdapat aplikasi pelaporan. Secara berkala, departemen teknologi informasi perusahaan akan menarik data dari beberapa basis data operasional dan menyajikannya dalam bentuk laporan kemudian selanjutnya diserahkan kepada level eksekutif perusahaan. Gambaran sistem *as-is* PT Mane Indonesia dapat dilihat pada gambar 3.2 Sistem *As-Is*



Gambar 3.2 Sistem *As-Is*

### 3.2.2 Teknologi Informasi di PT Mane Indonesia

Proses bisnis yang sedang berjalan di PT Mane Indonesia tidak lepas dari dukungan teknologi informasi baik perangkat keras maupun piranti lunak yang disediakan dan dipelihara oleh departemen *Information Technology* perusahaan.

#### 3.2.2.1 Perangkat Keras

Perangkat keras yang tersedia dan digunakan di PT Mane Indonesia dengan spesifikasi di bawah ini.

##### 1. Database dan Application Server

- Prosesor Intel ® Xeon ® CPU E5345 @ 2.33GHz
- Memori utama 4.00 GB
- Harddisk 132 GB
- DVD/CD drives TEAC DW-224E-V
- Network adapter HP NC373i Multifunction Gigabit Server Adapter, HP NC373i Multifunction Gigabit Server Adapter #2, HP Network Team #1

##### 2. Exchange/ Mail Server

- Prosesor Intel ® Xeon ™ CPU 3.60GHz
- Memori utama 2.00 GB
- Harddisk 204 GB
- DVD/CD-ROM TSSTcorp CD-ROM TS-L162C
- Network adapter HP NC7782 Gigabit Server Adapter, HP NC7782 Gigabit Server Adapter #2, HP Network Team #1

##### 3. Print Server

- Prosesor Intel ® Xeon ® CPU E5345 @ 2.33GHz
  - Memori utama 4.00 GB
  - Harddisk 68 GB
  - DVD/CD-ROM S-124
  - Network adapter HP NC7782 Server Adapter, HP NC7782 Server Adapter #2, HP Network Team #1
4. Switch
  5. Hub
  6. Work Station (spesifikasi work station yang dituliskan merupakan spesifikasi minimum yang terdapat di perusahaan)
    - Prosesor 1GHz
    - Memori utama 256 MB
    - Harddisk 40 GB
    - Mouse, monitor, dan keyboard sebagai peralatan antarmuka
  7. Printer yang terdapat di masing-masing departemen PT Mane Indonesia sejumlah 30 buah.

### **3.2.2.2 Piranti Lunak**

Piranti lunak yang digunakan di PT Mane Indonesia antara lain adalah Microsoft Windows 2003 Server, Microsoft Windows XP Profesional, dan Microsoft Windows 2000 untuk sistem operasi, Microsoft SQL Server 2000 untuk basis data operational, serta MS. Access 2000 dan Microsoft Visual Studio .NET 2003 untuk aplikasi *as-is*.

### 3.3 ANALISIS SWOT

Pengamatan terhadap faktor internal dan eksternal merupakan bagian dari *strategic planning process*. Faktor internal perusahaan biasanya dapat diklasifikasikan sebagai *strengths* (S) atau *weakness* (W), dan faktor eksternal dapat diklasifikasikan sebagai *opportunities* (O) atau *threat* (T). Analisa lingkungan strategis sedemikian rupa mengacu ke analisis SWOT.

Analisis SWOT digunakan untuk menyediakan informasi yang berguna untuk menyepadankan sumber daya dan kemampuan yang dimiliki oleh perusahaan untuk menghadapi lingkungan persaingan. Dengan kata lain analisis ini berfungsi untuk menyeleksi dan merumuskan strategi perusahaan. Berikut ini merupakan analisis SWOT lingkungan PT Mane Indonesia dalam kaitannya dengan teknologi informasi.

#### 1. Analisis Lingkungan Internal

- ***Strengths* (Kekuatan)**

Kemampuan dan sumber daya yang dimiliki perusahaan yang dapat digunakan sebagai dasar untuk mencapai kemajuan perusahaan. Berikut ini adalah kekuatan yang dimiliki PT Mane Indonesia yang dapat digunakan untuk berkembang dan bersaing dengan perusahaan lain

- a. Sumber daya manusia yang berkualitas

Keahlian dan kompetensi karyawan adalah faktor yang sangat penting dalam membangun kualitas sebuah perusahaan. PT Mane Indonesia memiliki kekuatan yang besar pada sumber daya manusia yang memiliki kemampuan



dalam menyelesaikan tugasnya, bertanggung jawab, ketepatan dalam menganalisa dan menyelesaikan masalah.

b. Menyediakan anggaran untuk pengadaan infrastruktur teknologi informasi  
PT Mane Indonesia menyadari bahwa teknologi sangat berperan penting dalam kelangsungan bisnis, sehingga anggaran untuk pengadaan infrastruktur teknologi informasi yang lebih baik selalu disediakan

c. Memiliki sistem informasi yang cukup handal

PT Mane Indonesia memiliki sistem informasi yang cukup handal dalam mendukung kegiatan operasionalnya yang dikembangkan oleh departemen teknologi informasi perusahaan.

- ***Weaknesses (Kelemahan)***

Kekurangan dari sumber daya dan kekuatan perusahaan dapat menjadi kelemahan. Beberapa kelemahan tersebut antarlain adalah:

a. Basis data operasional belum terintegrasi

Sistem informasi yang sedang berjalan di PT Indonesia didukung oleh beberapa basis data operasional yang belum terintegrasi. Hal ini mengakibatkan penyajian informasi tertentu kurang efisien karena harus mengakses beberapa sumber data yang terletak di lokasi yang berbeda.

b. *Analysis system* belum tersedia

Sistem informasi yang dimiliki PT Mane Indonesia sudah mampu dalam mendukung kegiatan transaksi harian perusahaan. Bagaimanapun juga seiring dengan semakin berkembangnya bisnis perusahaan maka data yang dikelola juga semakin banyak, dan pihak pada level tertentu membutuhkan sebuah

sistem yang mampu membantu mereka dalam menganalisa data sehingga dapat digunakan untuk mengambil keputusan. Sistem ini belum tersedia di PT Mane Indonesia.

## 2. Analisis Lingkungan Eksternal

- ***Opportunities (Peluang)***

Analisa terhadap lingkungan eksternal perusahaan dapat membuka peluang untuk meraih keuntungan dan perkembangan bisnis. Beberapa peluang tersebut antarlain adalah:

- a. Melakukan sinkronisasi basis data operasional ke dalam sebuah basis data analisis

Perkembangan teknologi informasi terutama dalam dunia basis data memungkinkan dilakukan transformasi dan sinkronisasi data yang terdapat pada basis data operasional ke dalam sebuah basis analisis.

- b. Mengembangkan *independent data mart*

*Independent data mart* merupakan bentuk sederhana *data warehouse* yang dapat mendukung proses pengambilan keputusan sebuah departemen dalam perusahaan.

Metode pengembangannya yang mirip dengan metode pengembangan *data warehouse* dapat menjawab masalah data yang tidak terintegrasi.

- c. Metodologi manajemen sistem basis data yang handal

Seiring dengan perkembangan sistem basis data, maka metodologi untuk perancangan, pengembangan, serta pemeliharaan basis data agar dapat menjadi sebuah basis data yang handal juga semakin berkembang dan mampu mengatasi masalah-masalah yang muncul pada sistem basis data.

- **Threats (Ancaman)**

Perubahan yang terjadi pada lingkungan eksternal dapat menjadi ancaman terhadap perusahaan. Beberapa ancaman tersebut antarlain adalah:

- a. Kelemahan *independent data mart*

*Independent data mart* memiliki beberapa kelemahan misalnya redundansi data karena konsep perancangannya yang terfokus pada kebutuhan sebuah departemen pada perusahaan.

- b. Harga pengadaan infrastruktur teknologi informasi yang cenderung terus meningkat

Perkembangan teknologi informasi tentu tidak lepas dari infrastruktur untuk mendukung operasinya. Semakin tinggi ketergantungan dengan teknologi maka infrastruktur yang dibutuhkan tentu harus dilengkapi dengan fitur-fitur yang semakin handal. Perkembangan fitur-fitur pada tools teknologi informasi mengakibatkan harga yang ditawarkan cenderung semakin meningkat.

Untuk mengembangkan strategi yang menggambarkan SWOT, maka faktor kekuatan, kelemahan, peluang, dan ancaman perusahaan tersebut dapat dituangkan ke dalam sebuah matriks. Matriks SWOT tersebut dapat dilihat pada halaman tabel (tabel 3. 1 Matriks SWOT PT Mane Indonesia).

Tabel 3.1 Matriks SWOT PT Mane Indonesia

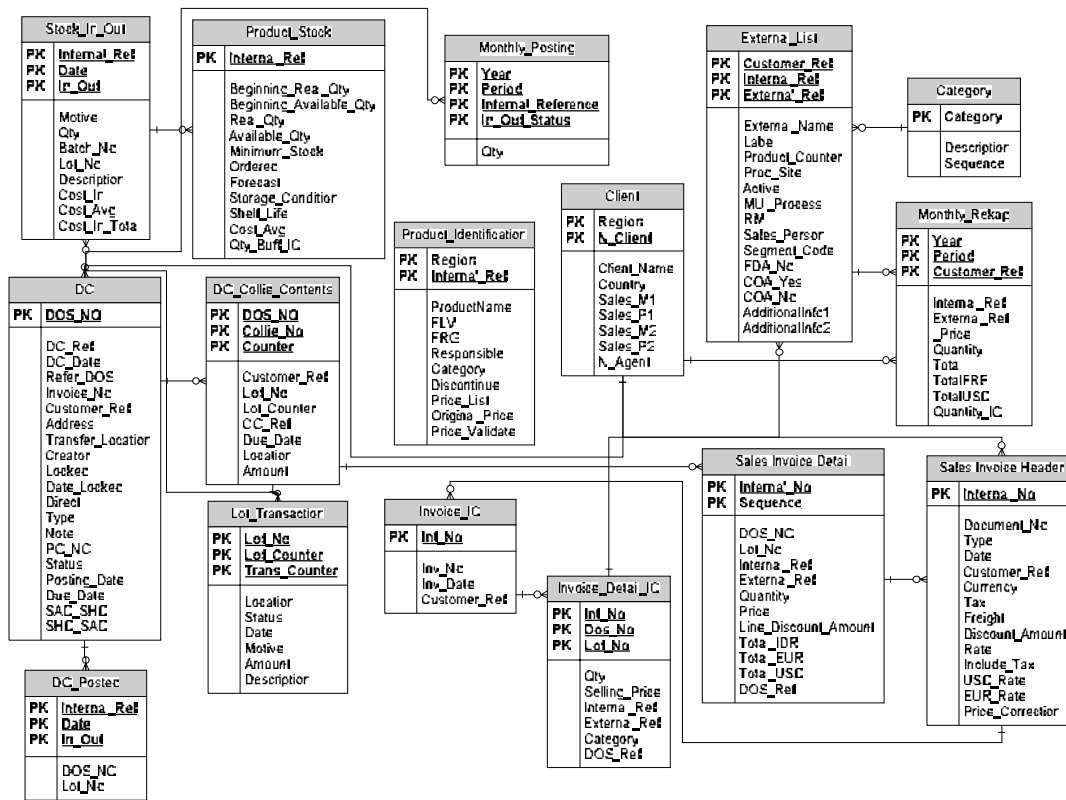
	<p><b>Strengths (S)</b></p> <ol style="list-style-type: none"> <li>1. Sumber daya manusia yang berkualitas</li> <li>2. Menyediakan anggaran untuk pengadaan infrastruktur teknologi informasi</li> <li>3. Memiliki Sistem informasi yang cukup handal</li> </ol>	<p><b>Weaknesses (W)</b></p> <ol style="list-style-type: none"> <li>1. Basis data operasional yang belum terintegrasi</li> <li>2. <i>Analysis system</i> belum tersedia</li> </ol>
<p><b>Opportunities (O)</b></p> <ol style="list-style-type: none"> <li>1. Melakukan sinkronisasi data dari basis data operasional ke dalam sebuah basis data analisis</li> <li>2. Mengembangkan <i>independent data mart</i></li> <li>3. Metodologi manajemen sistem basis data yang handal</li> </ol>	<p><b>Strategi SO</b></p> <ol style="list-style-type: none"> <li>1. Memanfaatkan SDM perusahaan terutama bagian teknologi informasi untuk melakukan sinkronisasi data dari basis data operasional ke dalam sebuah basis data analisis (S1,O1)</li> <li>2. Mengadakan infrastruktur untuk mengembangkan <i>independent data mart</i> (S3,O2)</li> </ol>	<p><b>Strategi WO</b></p> <ol style="list-style-type: none"> <li>1. Melakukan integrasi data dengan mengembangkan sebuah basis data analisis (W1,O1)</li> <li>2. Mengembangkan sebuah basis data analisis (W2,O2)</li> </ol>
<p><b>Threats (T)</b></p> <ol style="list-style-type: none"> <li>1. Kelemahan <i>independent data mart</i> (redundansi data)</li> <li>2. Harga pengadaan infrastruktur teknologi informasi yang cenderung terus meningkat</li> </ol>	<p><b>Strategi ST</b></p> <ol style="list-style-type: none"> <li>1. Menganalisa dan merancang pengembangan <i>independent data mart</i> dengan maksimal (S1,T1)</li> <li>2. Mengadakan anggaran yang maksimal untuk pengadaan infrastruktur teknologi informasi (S2,T2)</li> </ol>	<p><b>Strategi WT</b></p> <ol style="list-style-type: none"> <li>1. Mengoptimalkan metode pengambilan data dari basis data operasional untuk mengurangi masalah redundansi data (W1,T1)</li> </ol>

### **3.4 DESKRIPSI BASIS DATA AS-IS**

Basis data yang digunakan sebagai sumber data sistem *as-is Inventory* dan *Sales* adalah basis data MERP dan Navision Buffer. Basis data MERP adalah basis data yang menggabungkan data dari beberapa basis data operasional untuk menyimpan data *inventory control* dan penjualan barang yang mendukung sistem *as-is Inventory* dan *Sales*. Beberapa basis data operasional yang tergabung dalam basis data MERP antara lain berasal dari basis data AR\_SQL, EXIS, IC, PBookSQL, dan PSIS\_SQL. Sedangkan basis data Navision Buffer merupakan basis data yang menyimpan data *invoicing* yang mendukung sistem *as-is* bagian Sales.

#### **3.4.1 ERD (*Entity Relationship Diagram*)**

Basis data operasional yang terdapat pada sistem *as-is* terdiri atas beberapa tabel yang digunakan untuk mendukung sebagai *data store*. Relasi tabel- tabel pendukung modul *Inventory Control* dan *Sales* dijelaskan pada gambar 3.3 ERD *Inventory Control & Sales*.



Gambar 3.3 ERD Inventory Control & Sales

### 3.4.2 Deskripsi Tabel

Terdapat beberapa tabel pada basis data MERP yaitu tabel AR\_SQL.Monthly\_Rekap, EXIS.DO, EXIS.DO\_Posted, EXIS.DO\_Collie\_Contents, EXIS.Lot\_Transaction, IC.Invoice\_IC, IC.Invoice\_Detail\_IC, PBookSQL.Category, PBookSQL.External\_List, PSIS\_SQL.Stock\_In\_Out, PSIS\_SQL.Product\_Stock, PSIS\_SQL.MonthlyPosting\_CS, SysSQLClone.Client, dan SysSQLClone.Product\_Identification. Tabel tabel yang terlibat dari basis data Navision\_Buffer adalah tabel Navision\_Buffer.Sales Invoice Header dan Navision\_Buffer.Sales Invoice Detail.

- **Tabel PSIS\_SQL.Stock\_In\_out**

Identifikasi : Stock\_In\_Out

Deskripsi isi : Tabel yang mencatat transaksi pemasukan dan pengeluaran stock

Jenis : Tabel data transaksi

Primary key : Internal\_Ref, Date, In\_Out

**Tabel 3.2 Stock In Out**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Internal_Ref	Varchar (11)	Tidak	-	Kode Internal Produk
Date	Datetime (8)	Tidak	-	Tanggal transaksi
In_Out	Varchar (1)	Tidak	-	Tipe transaksi
Motive	Varchar (1)	Ya	-	Motive transaksi
Qty	Float (8)	Ya	(0)	Jumlah
Batch_No	Varchar (6)	Ya	-	No identitas barang untuk proses produksi
Lot_No	Varchar (8)	Ya	-	No identitas barang untuk penerimaan <i>stock</i> di gudang
Description	Varchar (100)	Ya	-	Keterangan proses keluar/ masuk <i>stock</i>
Cost_In	Binary (8)	Ya	(0)	Harga satuan barang masuk
Cost_Avg	Float (8)	Ya	(0)	Harga rata-rata barang
Cost_In_Total	Float (8)	Ya	(0)	Harga total barang masuk

- **Tabel PSIS\_SQL.MonthlyPosting\_CS**

Identifikasi : MonthlyPosting\_CS

Deskripsi isi : Tabel yang mencatat data posting *stock* setiap bulan

Jenis : Tabel data transaksi

Primary key : Internal\_Ref, Date, In\_Out

**Tabel 3.3 MontlyPosting\_CS**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Year	Varchar (9)	Tidak	-	Tahun posting
Period	Smallint (2)	Tidak	-	Bulan posting
Internal_Reference	Varchar (50)	Tidak	-	Kode Internal Produk
In_Out_Status	Float (8)	Ya	(0)	Status transaksi (keluar/ masuk)
Qty	Binary	Ya	-	Jumlah

- **Tabel PSIS\_SQL.Product\_Stock**

Identifikasi : Product\_Stock

Deskripsi isi : Tabel yang mencatat detail stock produk

Jenis : Tabel data induk

Primary key : Internal\_Ref

**Tabel 3.4 Product Stock**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Internal_Ref	Varchar (11)	Tidak	-	Kode internal barang
Beginning_Real_Qty	Float (8)	Ya	(0)	Jumlah awal barang yang tersedia
Beginning_Available_Qty	Float (8)	Ya	(0)	Jumlah awal barang yang siap digunakan
Real_Qty	Float (8)	Ya	(0)	Jumlah barang yang tersedia
Available_Qty	Float (8)	Ya	(0)	Jumlah barang yang siap digunakan
Minimum_Stock	Float (8)	Ya	(0)	Jumlah minimal stock yang harus tersedia
Ordered	Float (8)	Ya	(0)	Jumlah barang dipesan



**Tabel 3.4 Lanjutan-Product Stock**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Forecast	Float (8)	Ya	(0)	Perkiraan jumlah barang yang harus dipesan
Storage_Condition	Varchar (1)	Ya	-	Kondisi penyimpanan barang
Shelf_Life	Smallint (2)	Ya	-	Perkiraan waktu lamanya barang disimpan
Cost_Avg	Float (8)	Ya	(0)	Harga beli rata-rata

- **Tabel EXIS.DO**

Identifikasi : DO

Deskripsi isi : Tabel yang mencatat data *delivery item* kepada pelanggan

Jenis : Tabel data induk

Primary key : DOS\_NO

**Tabel 3.5 DO**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
DOS_NO	Varchar (6)	Tidak	-	No delivery order pada sistem
DO_Ref	Varchar (10)	Ya	-	No surat jalan ke pelanggan
DO_Date	Datetime (8)	Ya	-	Tanggal surat jalan
Refer_DOS	Varchar (6)	Ya	-	No referensi surat jalan (jika ada)
Invoice_No	Varchar (20)	Ya	-	No invoice
Customer_Ref	Int (4)	Ya	-	Kode pelanggan
Address	Varchar (200)	Ya	-	Alamat tujuan pengiriman barang
Transfer_Location	Varchar (5)	Ya	-	Lokasi transfer barang (berlaku untuk transfer stock antar lokasi barang)

Tabel 3.5 Lanjutan-DO

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Creator	Varchar (3)	Ya	-	Inisial pembuat surat jalan
Locked	Bit (1)	Tidak	(0)	Status <i>lock</i> surat jalan (tidak dapat dimodifikasi lagi)
Date_Locked	Datetime (8)	Ya	-	Tanggal pembuatan status <i>lock</i>
Direct	Bit (1)	Tidak	(0)	Pengiriman barang langsung ke pelanggan (tanpa perantara)
Type	Smallint (2)	Ya	-	Jenis surat jalan (domestic, export, transfer)
Note	Varchar (255)	Ya	-	Catatan pengiriman barang (jika ada)
PO_No	Varchar (255)	Ya	-	No <i>purchase order</i> dari pelanggan
Status	Smallint (2)	Ya	-	Status surat jalan ( <i>printed, posted, canceled</i> )
Posting_Date	Datetime (8)	Ya	-	Tanggal posting surat jalan
Due_Date	Datetime (8)	Ya	-	Tanggal batas waktu pengiriman barang
SAD_SHD	Bit (1)	Tidak	(0)	Status surat jalan sudah disiapkan oleh bagian <i>sales administration</i>
SHD_SAD	Bit (1)	Tidak	(0)	Status surat jalan sudah dilengkapi oleh bagian <i>shipping</i>

- **Tabel EXIS.DO\_Posted**

Identifikasi : DO\_Posted

Deskripsi isi : Tabel yang mencatat data surat jalan *delivered item* kepada pelanggan dengan status sudah *posting*

Jenis : Tabel data detail

Primary key : Internal\_ref, Date, In\_Out

**Tabel 3.6 DO\_Posted**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Internal_Ref	Varchar (11)	Tidak	-	Kode internal barang
Date	Datetime (8)	Tidak	-	Tanggal <i>posting</i> surat jalan
In_Out	Varchar (1)	Tidak	-	Status masuk/keluar <i>stock</i>
DOS_NO	Varchar (6)	Ya	-	No <i>delivery order</i>
Lot_No	Varchar (8)	Ya	-	No identitas barang untuk penerimaan <i>stock</i> di gudang

- **Tabel EXIS. DO\_Collie\_Contents**

Identifikasi : DO\_Collie\_Contents

Deskripsi isi : Tabel yang mencatat data detail barang yang tertera dalam surat jalan *delivery item* kepada pelanggan

Jenis : Table data detail

Primary key : DOS\_NO, Collie\_No, Counter

**Tabel 3.7 DO\_Collie\_Contents**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
DOS_NO	Varchar (6)	Tidak	-	No <i>delivery order</i>
Collie_No	Varchar (10)	Tidak	-	No muatan

**Tabel 3.7 Lanjutan-DO\_Collie\_Contents**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Counter	Int (4)	Tidak	-	
Customer_Ref	Int (4)	Ya	-	Kode pelanggan
Lot_No	Varchar (8)	Ya	-	No identitas barang untuk penerimaan <i>stock</i> di gudang
Lot_Counter	Int (4)	Ya	-	
CO_Ref	Varchar (15)	Ya	-	Kode order pelanggan
Due_Date	Datetime (8)	Ya	-	Tanggal batas waktu pengiriman barang
Location	Varchar (5)	Ya	-	Lokasi pengambilan stock
Amount	Smallint (2)	Ya	-	Jumlah barang

- **Tabel EXIS.Lot\_Transaction**

Identifikasi : Lot\_Transaction

Deskripsi isi : Tabel yang mencatat data transaksi lot barang

Jenis : Tabel data detail

Primary key : Lot\_no, Lot\_counter, Trans\_counter

**Tabel 3.8 Lot\_Transaction**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Lot_No	Varchar (8)	Tidak	-	No identitas barang untuk penerimaan <i>stock</i> di gudang
Trans_Counter	Int (4)	Tidak	-	No transaksi
Location	Varchar (5)	Ya	-	Lokasi penyimpanan stock
Status	Smallint (2)	Ya	-	Status transaksi lot
Date	Datetime (8)	Ya	-	Tanggal transaksi

**Tabel 3.8 Lanjutan-Lot\_Transaction**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Motive	Smallint (2)	Ya	-	Jenis transaksi lot ( <i>stock</i> bertambah, <i>delivery</i> kepada pelanggan, transfer lokasi)
Amount	Smallint (2)	Ya	-	Jumlah barang
Description	Varchar (100)	Ya	-	Keterangan transaksi

- **Tabel AR\_SQL.Monthly\_Rekap**

Identifikasi : Monthly\_Rekap

Deskripsi isi : Tabel yang mencatat data rekapitulasi penjualan barang per bulan

Jenis : Tabel data induk

Primary key : Year, Period, Customer\_Ref, Internal\_Ref, External\_Ref, Price

**Tabel 3.9 Monthly\_Rekap**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Year	Varchar (9)	Tidak	-	Tahun rekap
Period	Smallint (2)	Tidak	-	Bulan rekap
Customer_Ref	Int (4)	Tidak	-	Kode pelanggan
Internal_Ref	Varchar (11)	Tidak	-	Kode internal barang
External_Ref	Varchar (15)	Tidak	-	Kode eksternal barang (yang dijual kepada pelanggan)
Price	Float (8)	Tidak	-	Harga satuan barang
Quantity	Float (8)	Ya	-	Jumlah barang
Total	Float (8)	Ya	-	Total harga Rupiah
TotalFRF	Float (8)	Ya	-	Total harga EUR
TotalUSD	Float (8)	Ya	-	Total USD
Quantity_IO	Float (8)	Ya	-	Jumlah barang masuk/keluar

- **Tabel IC.Invoice\_IC**

Identifikasi : Invoice\_IC

Deskripsi isi : Tabel yang mencatat invoice dan barang yang dikirimkan

Jenis : Tabel data induk

Primary key : Int\_No

**Tabel 3.10 Invoice\_IC**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Int_No	Varchar (10)	Tidak	-	Kode internal
Inv_No	Varchar (20)	Ya	-	No invoice
Inv_Date	Datetime (8)	Ya	-	Tanggal invoice
Customer_Ref	Int (4)	Ya	-	Kode pelanggan

- **Tabel IC.Invoice\_Detail\_IC**

Identifikasi : Invoice\_Detail\_IC

Deskripsi isi : Tabel yang mencatat detail invoice dan barang yang dikirimkan

Jenis : Tabel data detail

Primary key : Int\_No, Dos\_No, Lot\_No

**Tabel 3.11 Invoice\_Detail\_IC**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Int_No	Varchar (10)	Tidak	-	Kode internal
Dos_No	Varchar (6)	Tidak	-	No <i>delivery order</i>
Lot_No	Varchar (8)	Tidak	-	No identitas barang untuk penerimaan <i>stock</i> di gudang
Qty	Float (8)	Ya	(0)	Jumlah barang
Selling_Price	Float (8)	Ya	(0)	Harga jual barang
Internal_Ref	Varchar (11)	Ya	-	Kode internal barang
External_Ref	Varchar (15)	Ya	-	Kode eksternal

Tabel 3.11 Lanjutan-Invoice\_Detail\_IC

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
				barang (yang dikirimkan kepada pelanggan)
Category	Varchar (4)	Ya	-	Kategori barang

- **Tabel Navision\_Buffer.Sales Invoice Header**

Identifikasi : Sales Invoice Header

Deskripsi isi : data *invoice* penjualan barang kepada pelanggan

Jenis : Tabel data induk

Primary key : Internal\_No

Tabel 3.12 Sales Invoice Header

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Internal_No	Varchar (10)	Tidak	-	Kode internal
Document_No	Varchar (20)	Ya	-	No dokumen
Type	Varchar (20)	Ya	-	Jenis dokumen ( <i>invoice/credit note/debit note</i> )
Date	Datetime (8)	Ya	-	Tanggal dokumen
Customer_Ref	Varchar (20)	Ya	-	Kode pelanggan
Currency	Varchar (10)	Ya	-	Mata uang
Tax	Money (8)	Ya	-	Jumlah pajak
Freight	Money (8)	Ya	-	Biaya angkutan
Discount_Amount	Money (8)	Ya	-	Jumlah potongan harga
Rate	Money (8)	Ya	-	Nilai tukar
Include_Tax	Int (4)	Ya	-	Status harga <i>invoice</i> termasuk pajak atau tidak
USD_Rate	Money (8)	Ya	-	Nilai tukar USD terhadap mata IDR
EUR_Rate	Money (8)	Ya	-	Nilai tukar EUR terhadap IDR

- **Tabel Navision\_Buffer.Sales Invoice Detail**

Identifikasi : Sales Invoice Detail

Deskripsi isi : Tabel yang mencatat data detail invoice penjualan barang kepada pelanggan

Jenis : Tabel data detail

Primary key : Internal\_No, Sequence

**Tabel 3.13 Sales Invoice Detail**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Internal_No	Varchar (10)	Tidak	-	Kode internal
DOS_NO	Varchar (20)	Ya	-	No <i>delivery order</i>
Lot_No	Varchar (20)	Ya	-	No identitas barang untuk penerimaan <i>stock</i> di gudang
Internal_Ref	Varchar (20)	Ya	-	Kode internal barang
External_Ref	Varchar (20)	Ya	-	Kode eksternal barang (yang dikirimkan kepada pelanggan)
Quantity	Money (8)	Ya	-	Jumlah barang
Price	Money (8)	Ya	-	Harga satuan barang
Line_Discount_Amount	Money (8)	Ya	-	Jumlah potongan harga
Total_IDR	Money (8)	Ya	-	Total harga dalam rupiah
Total_EUR	Money (8)	Ya	-	Total harga dalam EUR
Total_USD	Money (8)	Ya	-	Total harga dalam USD
DOS_Ref	Varchar (20)	Ya	-	No surat jalan



- **Tabel PBookSQL.External\_List**

Identifikasi : External\_List

Deskripsi isi : Tabel yang mencatat data barang per pelanggan yang akan dicantumkan pada surat jalan

Jenis : Tabel data induk

Primary key : Customer\_Ref, Internal\_Ref, External\_Ref

**Tabel 3.14 External\_List**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Customer_Ref	Int (4)	Tidak	-	Kode pelanggan
Internal_Ref	Varchar (11)	Tidak	-	Kode internal barang
External_Ref	Varchar (15)	Tidak	-	Kode eksternal barang (yang dikirimkan kepada pelanggan)
External_Name	Varchar (50)	Ya	-	Nama eksternal barang (yang dikirimkan kepada pelanggan)
Label	Varchar (50)	Ya	-	Label barang pada saat pengiriman
Category	Varchar (4)	Ya	-	Kategori barang
Product_Counter	Int (4)	Ya	-	
Prod_site	Int (4)	Ya	-	Inisial lokasi produksi barang
Active	Bit (1)	Ya	(0)	Status barang (masih dijual atau tidak)
MUI_Process	Bit (1)	Ya	(0)	Status barang apakah melalui proses MUI atau tidak
RM	Bit (1)	Ya	(0)	Jenis barang apakah tergolong bahan mentah atau tidak
Sales_Person	Varchar (10)	Ya	-	Kode <i>sales person</i>
Segment_Code	Varchar (5)	Ya	-	Kode bagian pemasaran
AdditionalInfo1	Varchar (20)	Ya	-	Informasi tambahan
AdditionalInfo2	Varchar (20)	Ya	-	Informasi tambahan

- **Tabel SysSQLClone.Product\_Identification**

Identifikasi : Product\_Identification

Deskripsi isi : Data barang

Jenis : Tabel data induk

Primary key : Region, Reference

**Tabel 3.15 Product\_Identification**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Region	Varchar (3)	Tidak	-	Kode wilayah
Reference	Varchar (11)	Tidak	-	Kode internal produk
ProductName	nVarchar (60)	Ya	-	Nama internal barang
FLV	Bit (1)	Ya	-	Divisi <i>flavour</i>
FRG	Bit (1)	Ya	-	Divisi <i>fragrance</i>
Responsible	nVarchar (3)	Ya	-	Divisi penanggung jawab
Category	nVarchar (2)	Ya	-	Kategori produk
Discontinue	Bit (1)	Ya	-	Status produk masih aktif atau tidak
Price_List	Float (8)	Ya	-	Harga jual minimum
Original_Price	Float (8)	Ya	-	Harga baku
Price_Validate	Datetime (8)	Ya	-	Batas valid harga
Halal	Bit (1)	Ya	-	Status halal
Origin	Int (4)	Ya	-	Lokasi asal produk
Supplier	Int (4)	Ya	-	<i>Supplier</i> produk
Technical_Comments	nVarchar (1000)	Ya	-	Catatan teknis

- **Tabel PBookSQL. Category**

Identifikasi : Category

Deskripsi isi : Data kategori barang

Jenis : Tabel data induk

Primary key : Category

**Tabel 3.16 Category**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Category	Varchar (10)	Tidak	-	Kode kategori barang
Description	Varchar (50)	Ya	-	Keterangan kode kategori
Sequence	Tinyint (1)	Ya	-	Urutan
[Group]	Tinyint (1)	Ya	-	Kelompok kategori barang

- **SysSQLClone.Client**

Identifikasi : Client

Deskripsi isi : Data pelanggan

Jenis : Tabel data induk

Primary key : Region, N\_Client

**Tabel 3.17 Client**

Id Field	Tipe & Ukuran	Boleh Null	Default	Deskripsi
Region	varchar (3)	Tidak	-	Kode wilayah
N_Client	int (4)	Tidak	-	Kode pelanggan
ClientName	nvarchar (80)	Ya	-	Nama pelanggan
Country	int (4)	Ya	-	Kode negara
Sales_M1	varchar (10)	Ya	-	Kode <i>sales manager</i> 1
Sales_P1	varchar (10)	Ya	-	Kode <i>sales person</i> 1

**Tabel 3.17 Lanjutan-Client**

<b>Id Field</b>	<b>Tipe &amp; Ukuran</b>	<b>Boleh Null</b>	<b>Default</b>	<b>Deskripsi</b>
Sales_M2	varchar (10)	Ya	-	Kode <i>sales manager 2</i>
Sales_P2	varchar (10)	Ya	-	Kode <i>sales person 2</i>
N_Agent	int (4)	Ya	-	Kode agen
OtherInformation	text (16)	Ya	-	Informasi tambahan

### 3.5 ANALISIS KEBUTUHAN DATA DAN INFORMASI DI PT MANE INDONESIA

Data dan informasi yang dihasilkan bagian *Inventory Control* dan *Sales* di PT Mane Indonesia merupakan data yang sangat penting yang mempengaruhi pengambilan keputusan.

#### 1. Data *Inventory Control*

Data yang terdapat pada bagian *Inventory Control* merupakan data pemasukan *stock* (hasil proses produksi, penerimaan barang dari *supplier*, hasil restorasi barang, pengembalian barang dari pelanggan) dan pengeluaran *stock* (proses produksi barang jadi, pengambilan barang/uji coba untuk *quality control*, pengambilan barang untuk laboratorium, pengiriman barang ke pelanggan).

Data dan informasi yang dibutuhkan dari bagian *Inventory Control* adalah:

- Jumlah *stock* masuk dan keluar berdasarkan kategori produk
- Jumlah barang yang digunakan untuk QC *retain*
- Jumlah barang yang digunakan untuk *sample* produksi di bagian Lab
- *Summary transaction* stock setiap bulan
- Jumlah dan harga barang yang dikirimkan kepada pelanggan berdasarkan periode *posting*

#### 2. Data *Sales*

Data pada bagian *Sales* merupakan data hasil pencatatan order dan pengiriman barang kepada pelanggan.

Data dan informasi yang dibutuhkan dari bagian *Sales* adalah:

- Data penjualan berdasarkan *sales person*, area, dan kategori produk yang dijual untuk periode tertentu baik secara bulanan maupun secara kumulatif tahunan
- Data penjualan berdasarkan agen pada masing-masing area penjualan baik secara bulanan maupun secara kumulatif tahunan
- Statistik penjualan berdasarkan kode produk
- Statistik penjualan berdasarkan kode pelanggan

### 3.6 MASALAH YANG DIHADAPI

PT Mane Indonesia merupakan salah satu perusahaan yang bergerak di bidang industri yang menghasilkan bahan dasar perasa makanan dan pewangi yang semakin berkembang sesuai dengan semakin meningkatnya permintaan pasar. Perkembangan bisnis ini tentunya diiringi dengan semakin banyaknya data dan informasi yang harus diolah secara optimal untuk membantu pihak manajemen perusahaan dalam mengambil keputusan yang cepat dan tepat. Dalam sistem yang berjalan sekarang ini terdapat beberapa masalah dalam mengolah data menjadi informasi yang dapat mendukung pengambilan keputusan terutama pada bagian *Sales* dan *Inventory Control*:

1. Basis data yang belum terintegrasi menyebabkan proses penarikan data untuk disajikan sebagai informasi yang berguna bagi pihak manajemen tidak efisien karena memerlukan waktu yang relatif lama
2. Permintaan *query* dari *internal user* yang cukup tinggi

Permintaan *query* data yang cukup sering dan berubah-ubah, cukup menyita waktu dan mempengaruhi efisiensi kerja departemen teknologi informasi selaku pemelihara basis data perusahaan. Sehingga diperlukan sebuah aplikasi yang dapat membantu *internal user* untuk melakukan *query* data sendiri.

3. Jumlah transaksi pemasukan dan pengeluaran *stock* yang tinggi.

Untuk membantu pihak manajemen perusahaan untuk mengambil keputusan, diperlukan aplikasi pelaporan atau penampilan data pemasukan dan pengeluaran *stock* yang saat ini belum tersedia.

4. Fluktuasi angka penjualan barang

Untuk membantu pihak manajemen perusahaan dalam melihat peningkatan ataupun penurunan angka penjualan dari berbagai dimensi dalam bentuk pelaporan belum tersedia

### 3.7 ALTERNATIF PEMECAHAN MASALAH

Setelah melakukan analisis terhadap permasalahan yang dihadapi dan kebutuhan informasi pada bagian *Inventory Control* dan *Sales* PT Mane Indonesia, maka diperoleh sebuah alternatif pemecahan masalah yaitu pengembangan sebuah basis data analisis berupa *independent data mart*.

*Independent data mart* merupakan sebuah basis data analisis yang dapat mendukung pengambilan keputusan dan fokus terhadap pada area fungsional tertentu dalam sebuah perusahaan.

*Independent data mart* dapat menyimpan data *histories* perusahaan sehingga data yang tersedia dapat dianalisa dan digunakan untuk membangun rencana strategis masa depan. *Independent data mart* juga dapat mengintegrasikan data dari beberapa basis data operasional dalam perusahaan, dengan cara memilih data yang dibutuhkan untuk menjawab kebutuhan informasi pada area fungsional tertentu, kemudian melakukan transformasi (*data cleaning* dan re-format) agar data yang disaring benar-benar merupakan data yang diperlukan untuk menjawab kebutuhan pengguna. Kemudian data yang sudah mengalami transformasi ini diisi ke dalam *independent data mart*.

Selanjutnya untuk menjamin akurasi data yang terdapat pada *independent data mart*, maka dilakukan sinkronisasi data dari basis data operasional. Metode sinkronisasi ini diatur sedemikian rupa sehingga eksekusi nya berjalan secara otomatis sesuai dengan jadwal yang akan ditetapkan tanpa harus dieksekusi secara manual.

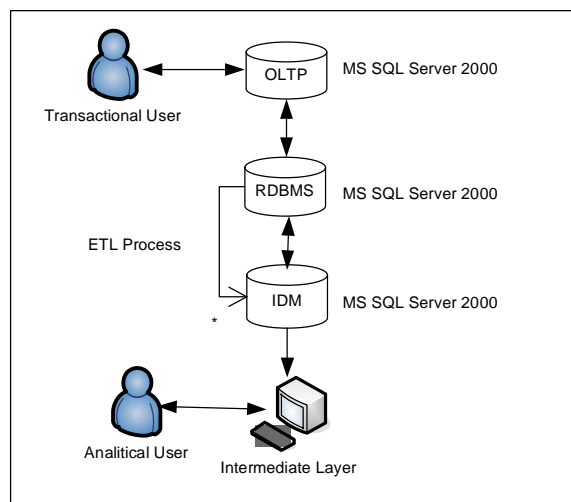


Akses informasi yang dihasilkan sebagai hasil pengolahan data yang terdapat pada *independent data mart*, dapat dilakukan melalui sebuah *intermediate layer* yang menghubungkan pengguna dengan *independent data mart*.

**BAB IV**  
**SINKRONISASI BASIS DATA OPERASIONAL**  
**MENJADI INDEPENDENT DATA MART**

**4.2 ARSITEKTUR INDEPENDENT DATA MART**

Arsitektur *independent data mart* secara umum dibagi menjadi dua bagian besar. Bagian pertama merupakan bagian dari arsitektur basis data operasional, menggunakan RDBMS sebagai media penyimpanan data operasional. Bagian berikutnya merupakan bagian dari sistem *independent data mart* itu sendiri, dimana sumber data diambil dari RDBMS kemudian dilakukan proses ETL. Multidimensional data yang telah dihasilkan akan diakses menggunakan aplikasi berbasis web yang merupakan *intermediate layer* antara pengguna dengan *independent data mart*.



**Gambar 4.1 Arsitektur Independent Data Mart**

Komponen-komponen pendukung yang terdapat pada aplikasi independent data mart PT Mane Indonesia antarlain:

#### 1. Data Operasional

Data operasional merupakan data yang berasal dari basis data yang digunakan PT Mane Indonesia dalam melakukan kegiatan operasional. Data operasional yang digunakan pada penulisan tugas akhir ini adalah data yang berhubungan dengan bagian *Inventory Control* dan *Sales* PT Mane Indonesia yang dibutuhkan untuk penyelesaian masalah yang terdapat pada bagian tersebut. Data yang terlibat yaitu:

- *Stock\_In\_Out*: tabel yang berisi data transaksi masuk dan keluar *stock* barang
- *Monthly\_Posting\_CS*: tabel yang berisi data posting *stock* barang setiap bulan
- *Product\_Stock*: tabel yang berisi detail *stock* barang
- *DO*: tabel yang berisi data *delivery order* ke pelanggan
- *DO\_Collie\_Contents*: tabel yang berisi detail *delivery order* ke pelanggan
- *Lot\_Transaction*: tabel yang berisi data transaksi lot barang
- *Monthly\_Rekap*: tabel yang berisi data rekapitulasi penjualan barang ke pelanggan
- *Invoice\_Detail\_IC*: tabel yang berisi data pemetaan invoice dengan stock
- *Sales Invoice Header*: tabel yang berisi data invoice yang dikirimkan kepada pelanggan
- *Sales Invoice Detail*: tabel yang berisi detail data invoice yang dikirimkan kepada pelanggan

- *External\_List*: tabel yang mencatat data barang per pelanggan yang akan dicantumkan pada surat jalan
- *Product\_Identification*: tabel yang berisi identifikasi produk
- *Category*: tabel yang berisi kategori produk
- *Client*: tabel yang berisi data pelanggan

Data operasional ini digunakan untuk menggambarkan ERD untuk kemudian digunakan untuk menemukan data yang relevan untuk subjek *Inventory Control* dan *Sales* kemudian akan dibangun skema bintang.

## 2. Proses ETL (*Extraction, Transformation, Loading*)

Setelah mengidentifikasi data dari basis data operasional, maka akan dilakukan proses ETL (*Extraction, Transformation, Loading*). Pada tahap *extraction* dilakukan proses penyeleksian data-data dan informasi yang relevan bagi keperluan analisis dan pengambilan keputusan dari beberapa sumber data dan platform yang berbeda. Transformasi dilakukan sebagai tahapan untuk transformasi dan integrasi skema serta struktur yang berbeda-beda ke dalam skema yang terstruktur yang terdefinisi dalam *independent data mart*. Loading merupakan proses pemindahan data secara fisik dari sistem operasional ke dalam *data warehouse*.

## 3. *Independent Data Mart*

*Independent Data Mart* adalah bentuk sederhana *data warehouse* yang terfokus pada satu pokok (satu bagian fungsional) seperti *Sales*, *Finance*, atau *Marketing*. *Independent Data Mart* biasanya dibangun dan diawasi oleh satu bagian dalam sebuah perusahaan atau organisasi.

#### 4. *Analitical user*

Pengguna aplikasi *independent data mart* yang merupakan pihak penentu keputusan pada bagian *inventory control* dan *sales*, serta pihak manajemen perusahaan untuk membantu proses pengambilan keputusan, yang disajikan dalam bentuk laporan.

### 4.3 PERANCANGAN INDEPENDENT DATA MART

Pada tahap perancangan, dilakukan beberapa tahapan:

1. Analisis permasalahan yang dihadapi bagian *Inventory Control* dan *Sales* PT Mane Indonesia serta kebutuhan informasi yang dibutuhkan
2. Klasifikasi Data untuk Skema Independent Data Mart

Pada tahapan ini dilakukan analisis terhadap skema data operasional dan data yang dibutuhkan secara umum oleh *independent data mart*. Kemudian menentukan dimensi, serta menentukan tabel fakta.

3. Membuat metadata mengenai deskripsi tabel fakta dan tabel dimensi yang terdapat pada skema bintang
4. Menentukan spesifikasi perangkat keras serta piranti lunak yang akan digunakan untuk menjalankan aplikasi *independent data mart*

#### 4.3.1 Permasalahan dan Kebutuhan Informasi

Seperti yang dijelaskan pada sub bab 3.6 bahwa ada beberapa masalah yang dihadapi oleh bagian *Inventory Control* dan *Sales* PT Mane Indonesia. Masalah tersebut antarlain menyangkut kebutuhan *query*, informasi yang ditampilkan dalam bentuk laporan untuk membantu pihak manajemen dalam mengambil keputusan, serta basis

data operasional yang belum terintegrasi sehingga mempengaruhi efisiensi penarikan data yang akan digunakan dalam penyajian informasi yang berguna untuk mendukung proses pengambilan keputusan.

#### **4.3.2 Kebutuhan Data Bagian *Inventory Control* dan *Sales***

Di PT Mane Indonesia bagian *Inventory Control* memiliki kebutuhan *query* yang berkaitan dengan penerimaan *stock* ke gudang dan distribusinya. Kebutuhan *query* bagian *Inventory Control* adalah:

1. Jumlah *raw material* yang tersedia di gudang
2. Jumlah *finished product* sebagai hasil produksi yang siap untuk disalurkan
3. Jumlah barang masuk dan keluar berdasarkan kategori produk
4. Jumlah barang yang digunakan untuk *QC retain*
5. Jumlah barang yang digunakan untuk *sample* produksi di bagian Lab
6. Jumlah *raw material* yang digunakan untuk produksi
7. Jumlah dan harga barang yang dikirimkan kepada pelanggan berdasarkan periode posting

Bagian *Sales* memiliki kebutuhan *query* yang berkaitan dengan penjualan produk dan pemasarannya. Kebutuhan bagian *Sales* adalah:

1. Data penjualan berdasarkan sales person
2. Data penjualan berdasarkan wilayah penjualan
3. Data penjualan berdasarkan kategori produk
4. Data penjualan berdasarkan agen pada masing-masing area penjualan baik secara bulanan

## 5. Statistik penjualan berdasarkan kategori produk

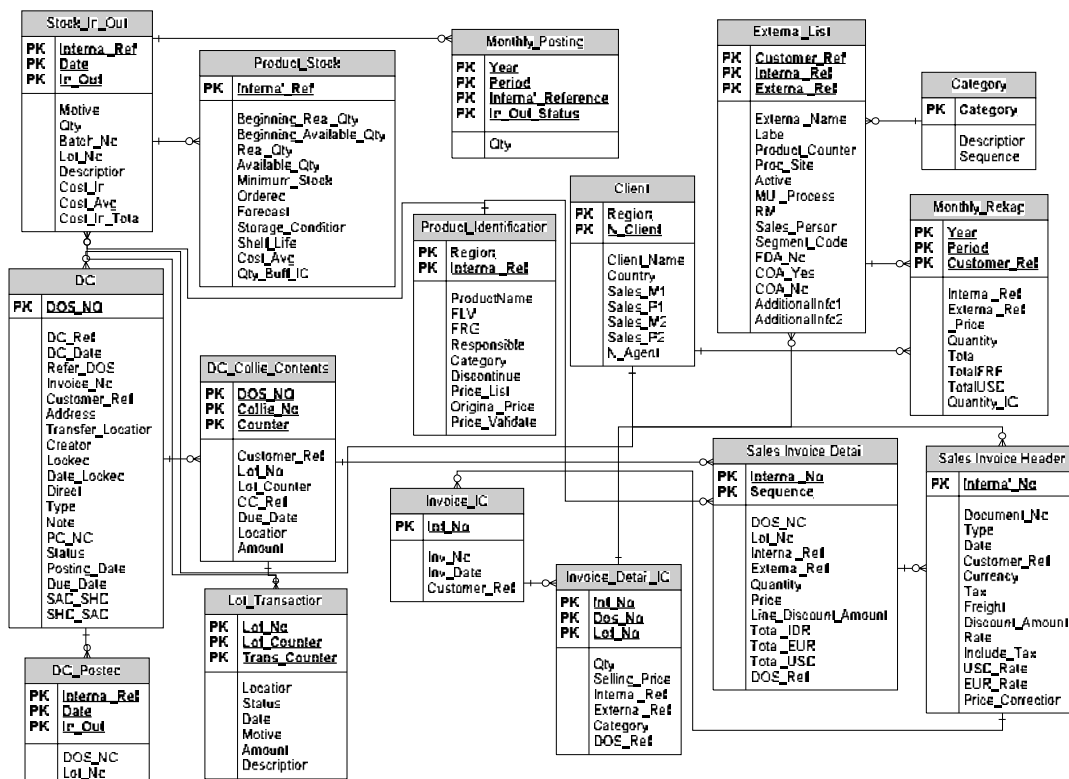
Statistik penjualan berdasarkan pelanggan

### 4.3.3 Klasifikasi Data Untuk Skema Independent Data Mart

Pada tahapan ini dilakukan klasifikasi data berupa pemilihan proses, *granularity*, fakta, dan dimensi. Gambaran umum fakta, dimensi, serta relasi nya dalam aplikasi *independent data mart* nantinya akan dituangkan dalam skema bintang.

#### 4.2.3.1 Skema Umum Basis Data Operasional

Skema operasional yang digunakan oleh PT Mane Indonesia merupakan skema relasional seperti pada gambar 4.2 Skema Basis Data Relasional *Inventory Control & Sales*



Gambar 4.2 Skema Basis Data Relasional Inventory Control & Sales

Skema di atas akan digunakan baik oleh bagian *Inventory Control* dan *Sales* dalam pembangunan *independent data mart*.

#### **4.2.3.2 Perancangan Skema Bintang**

Pada tahapan perancangan skema bintang dilakukan empat tahapan yaitu pemilihan proses bisnis yang akan dimodelkan, pemilihan grain dari proses bisnis, membuat dimensi, dan membuat tabel fakta yang akan dijadikan sebagai komponen skema bintang.

##### **4.2.3.2.1 Memilih Proses**

Sebuah proses adalah kegiatan alamiah bisnis yang ada pada perusahaan. Proses bisnis yang dipilih pada bagian *Inventory control* dan *Sales* PT Mane Indonesia adalah:

- Transaksi Stock
- Transaksi QC Retain
- Transaksi Lab Delivery
- Transaksi Stock In Out
- Transaksi Shipment Posting
- Sales Realised
- Transaksi Penjualan
- Statistik Penjualan



#### 4.2.3.2.2 Memilih Granularity

Memilih grain berarti menetapkan apa yang akan dijelaskan pada setiap baris tabel fakta. Grain dari bisnis proses yang telah dipilih adalah:

- Transaksi Stock

Grain pada fakta *stock* adalah variabel yang mewakili fakta *stock* berupa jumlah dan harga satuan barang

- Transaksi QC Retain

Grain pada fakta QC Retain adalah variabel yang mewakili fakta QC Retain berupa jumlah dan harga satuan barang

- Transaksi Lab Delivery

Grain pada fakta Lab Delivery adalah variabel yang mewakili fakta Lab Delivery berupa jumlah dan harga satuan barang

- Transaksi Stock In Out

Grain pada fakta Stock In Out adalah variabel yang mewakili fakta Stock In Out berupa jumlah, harga satuan barang keluar atau masuk

- Transaksi Shipment Posting

Grain pada fakta Shipment Posting adalah variabel yang mewakili fakta Shipment Posting berupa jumlah, harga satuan tertagih, dan biaya produksi

- Sales Realised

Grain pada fakta Sales Realised adalah variabel yang mewakili Sales Realised berupa total penjualan dan *product site*

- Transaksi Penjualan

Grain pada fakta Penjualan adalah variabel yang mewakili fakta Penjualan berupa total penjualan

- Statistik Penjualan

Grain pada fakta Statistik Penjualan adalah variabel yang mewakili fakta Statistik Penjualan berupa jumlah, harga penjualan, total IDR, total USD, total EUR

Level granularity yang akan ditampilkan pada aplikasi independent data mart Inventory Control dan Sales PT Mane Indonesia adalah:

- Data stock serta harga satuan per hari, bulan, tahun berdasarkan kategori barang
- Data stock in out barang serta harga satuan per hari, bulan, tahun berdasarkan jenis transaksi dan jenis barang
- Data jumlah QC Retain serta harga satuan per bulan berdasarkan divisi dan jenis barang
- Data jumlah Lab Delivery serta harga satuan per bulan berdasarkan divisi dan jenis barang
- Data jumlah shipment posting serta harga satuan tertagih per bulan berdasarkan pelanggan dan mata uang
- Data jumlah transaksi serta harga penjualan per bulan berdasarkan motif transaksi dan status penjualan
- Data jumlah penjualan serta *production site* per bulan berdasarkan *sales person*, negara, dan pelanggan

- Data jumlah serta total angka penjualan per bulan, tahun, *month to date*, *year to date* berdasarkan *sales person*, kategori barang, area, dan pelanggan
- Data statistik jumlah serta total angka penjualan berdasarkan pelanggan, produk, negara (lokal atau ekspor), dan mata uang

#### 4.2.3.2.3 Dimensi

Pada tahap ini dipilih dimensi yang akan dipakai pada setiap row tabel fakta. Dimensi dibuat untuk menjawab pertanyaan bagaimana kalangan bisnis menjelaskan data yang dihasilkan dari bisnis proses. Setelah memahami grain yang dipilih maka pembuatan dimensi akan relatif mudah.

Berikut adalah daftar dimensi yang akan dibentuk pada independent data mart *Inventory Control* dan *Sales PT Mane Indonesia*.

**Tabel 4.1 Dimensi**

Dimensi	Level Dimensi	Dasar Perancangan
Period	<ul style="list-style-type: none"> <li>• Tahun</li> </ul>	Dibutuhkan oleh semua fact pada bagian Inventory Control dan Sales
	<ul style="list-style-type: none"> <li>• Bulan</li> </ul>	Dibutuhkan oleh semua fact pada bagian Inventory dan Sales
	<ul style="list-style-type: none"> <li>• Hari</li> </ul>	Dibutuhkan oleh fact stock dan stock in out pada bagian Inventory Control dan Sales
	<ul style="list-style-type: none"> <li>• Month to date</li> </ul>	Dibutuhkan oleh fact Penjualan dan Statistik Penjualan pada bagian Sales
	<ul style="list-style-type: none"> <li>• Year to date</li> </ul>	Dibutuhkan oleh fact Penjualan dan Statistik Penjualan pada bagian Sales
Product	<ul style="list-style-type: none"> <li>• Kode Produk</li> </ul>	Dibutuhkan oleh fact Stock, Stock In Out, QC Retain, dan Lab Delivery pada bagian Inventory Control dan fact Sales Realised dan Statistik

Tabel 4.1 Lanjutan-Dimensi

Dimensi	Level Dimensi	Dasar Perancangan
	<ul style="list-style-type: none"> <li>Nama Produk</li> </ul>	Dibutuhkan oleh fact Stock, Stock In Out, QC Retain, dan Lab Delivery pada bagian Inventory Control dan fact Sales Realised dan Statistik Penjualan pada bagian Sales
Division	<ul style="list-style-type: none"> <li>Divisi</li> </ul>	Dibutuhkan oleh fact QC Retain dan Lab Delivery bagian Inventory Control
Customer	<ul style="list-style-type: none"> <li>Kode Pelanggan</li> </ul>	Dibutuhkan oleh fact Shipment Posting pada bagian Inventory Control dan semua fact pada bagian Sales
	<ul style="list-style-type: none"> <li>Nama Pelanggan</li> </ul>	Dibutuhkan oleh fact Shipment Posting pada bagian Inventory Control dan semua fact pada bagian Sales
Motive	<ul style="list-style-type: none"> <li>Motif Transaksi</li> </ul>	Dibutuhkan oleh fact Stock In Out dan Summarize Transaction pada bagian Inventory Control
Category Jual	<ul style="list-style-type: none"> <li>Kategori Produk Dijual</li> </ul>	Dibutuhkan oleh fact Summarize Transaction pada bagian Inventory Control dan fact Sales Realised pada bagian Sales
Currency	<ul style="list-style-type: none"> <li>Currency</li> </ul>	Dibutuhkan oleh fact Shipment Posting pada bagian Inventory Control dan fact Statistik Penjualan pada bagian Sales
Country	<ul style="list-style-type: none"> <li>Country</li> </ul>	Dibutuhkan fact Sales Realised pada bagian Sales
Sales	<ul style="list-style-type: none"> <li>Sales Person Code</li> </ul>	Dibutuhkan oleh fact Shipment Posting pada bagian Inventory Control dan semua fact pada bagian Sales
Area	<ul style="list-style-type: none"> <li>Area Penjualan</li> </ul>	Dibutuhkan oleh fact Penjualan pada bagian Sales
State	<ul style="list-style-type: none"> <li>State</li> </ul>	Dibutuhkan oleh fact Summarize Transaction pada bagian Inventory Control dan fact Statistik Penjualan pada bagian Sales

#### 4.2.3.2.4 Fakta

Fakta adalah data *item* yang numerik yang merepresentasikan kepentingan bisnis, digunakan oleh *user* untuk menganalisis dan menarik kesimpulan agar mendapatkan pengertian yang lebih baik akan bisnis yang ada.

**Tabel 4.2 Fakta**

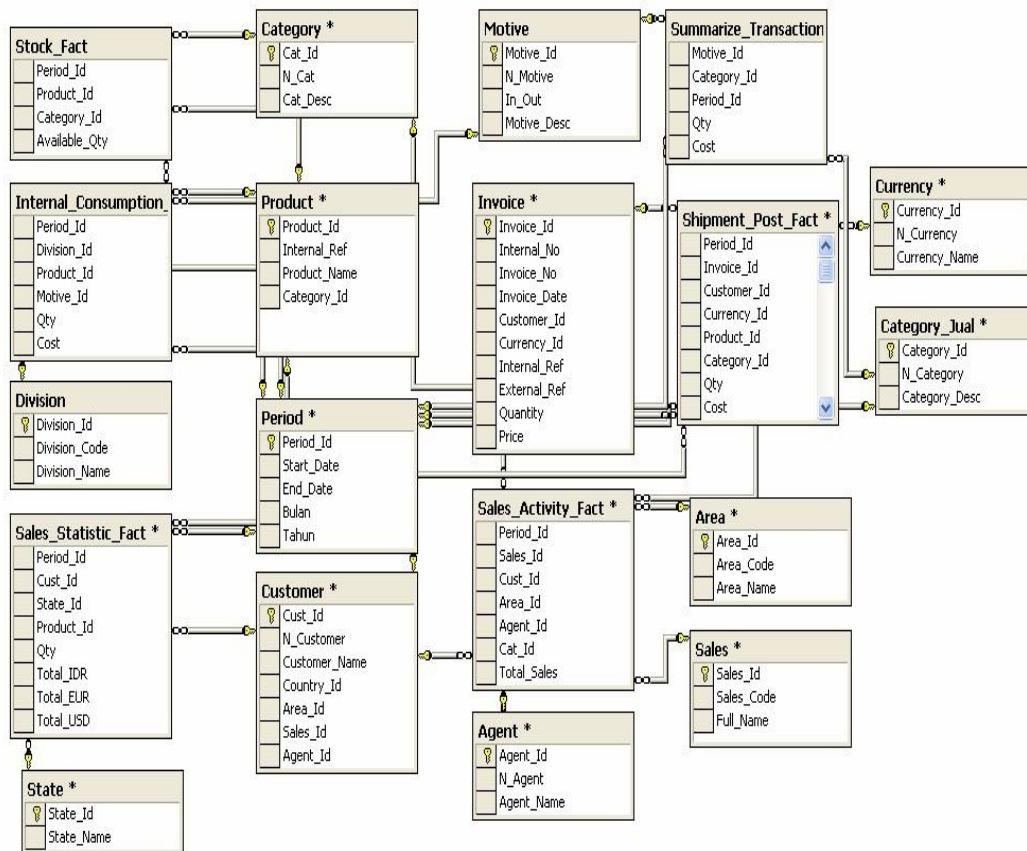
Tabel Fakta	Tabel yang dipetakan dari data source	Dasar Perancangan
Stock_Fact	Stock_In_Out	Subject area adalah stock
Internal_Consumption_Fact	Stock_In_Out	
Shipment_Post_Fact	<ul style="list-style-type: none"> <li>• Stock_In_Out</li> <li>• Invoice_IC</li> <li>• Invoice_Detail_IC</li> <li>• DO_Posted</li> </ul>	Subjet area adalah shipment barang
Summarize_Transaction_Fact	Stock_In_Out	Subject area adalah transaksi stok by motive
Sales_Activity_Fact	Monthly_Rekap	Subject area adalah Penjualan
Sales_Statistic_Fact	Monthly_Rekap	Subject area adalah statistik penjualan

#### 4.2.3.2.5 Skema Bintang

Skema bintang menggambarkan secara umum relasi dimensi dan fakta yang telah dibuat. Rancangan skema bintang Independent Data Mart PT Mane Indonesia bagian Inventory Control dan Sales akan digambarkan secara umum. Kemudian skema bintang masing-masing bagian akan dijelaskan secara rinci.

### 1. Skema Bintang Gabungan

Skema bintang gabungan Inventory Control dan Sales PT Mane Indonesia dapat dilihat pada gambar 4.3 Skema Bintang Gabungan

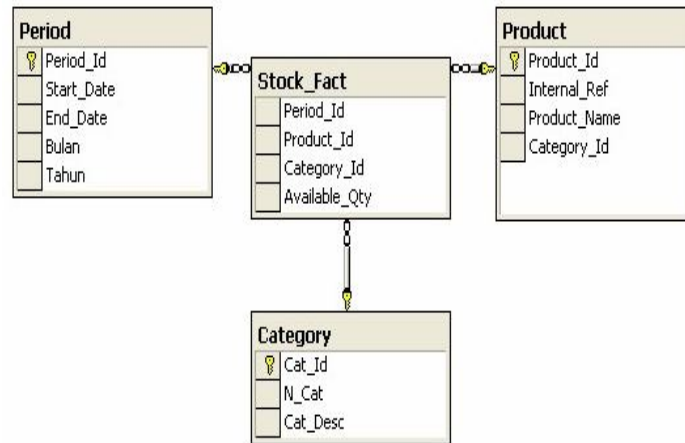


Gambar 4.3 Skema Bintang Gabungan

## 2. Skema Bintang Stock

Skema bintang stock merupakan model dimensional penyajian informasi stock pada bagian *Inventory Control*.

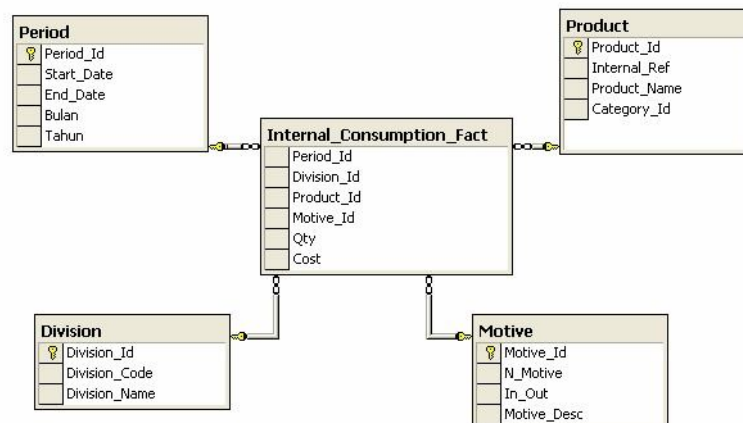
Skema bintang stock terdiri dari 1 tabel fakta dan 3 tabel dimensi. Tabel fakta yaitu *Stock\_Fact* dan tabel dimensi yaitu *Period*, *Product*, dan *Category*.



Gambar 4.4 Skema Bintang Stock

### 3. Skema Bintang Internal\_Consumption

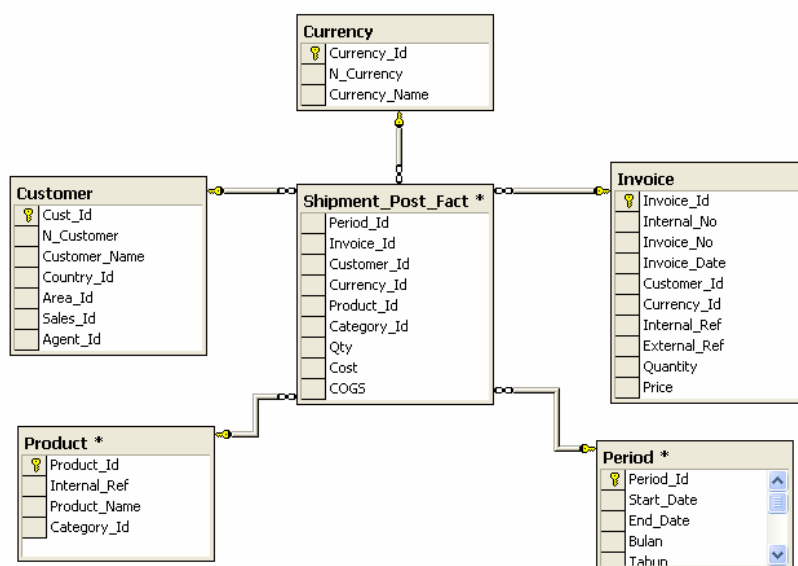
Skema Bintang Internal\_Consumption merupakan model dimensional penyajian informasi konsumsi internal stock pada bagian *Inventory Control*. Skema Internal\_Consumption terdiri dari 1 tabel fakta dan 4 tabel dimensi. Tabel fakta yaitu Internal\_Consumption\_Fact dan tabel dimensi yaitu Period, Product, Division, dan Motive.



Gambar 4.5 Skema Bintang Internal\_Consumption

#### 4. Skema Bintang Shipment\_Posting

Skema bintang Shipment\_Posting merupakan model dimensional penyajian informasi pengiriman barang dengan *invoice*. Skema bintang Shipment\_Posting terdiri dari 1 tabel fakta dan 5 tabel dimensi. Tabel fakta yaitu Shipment\_Post\_Fact dan tabel dimensi yaitu Currency, Customer, Invoice, Product, Period.



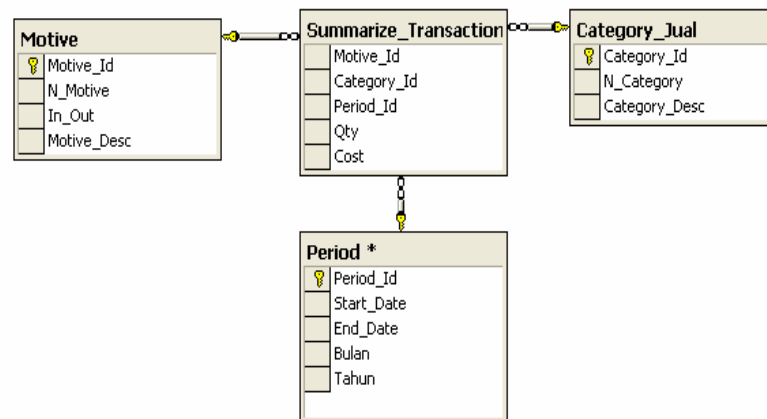
Gambar 4.6 Skema Bintang Shipment\_Posting

#### 5. Skema Bintang Summarize\_Transaction

Skema bintang Summarize\_Transaction merupakan model dimensional penyajian informasi rangkuman transaksi penggunaan stock.



Skema bintang Summarize\_Transaction terdiri dari 1 tabel fakta dan 3 tabel dimensi. Tabel fakta yaitu Summarize\_Transaction dan tabel dimensi yaitu Category\_Jual, Motive, dan Period.

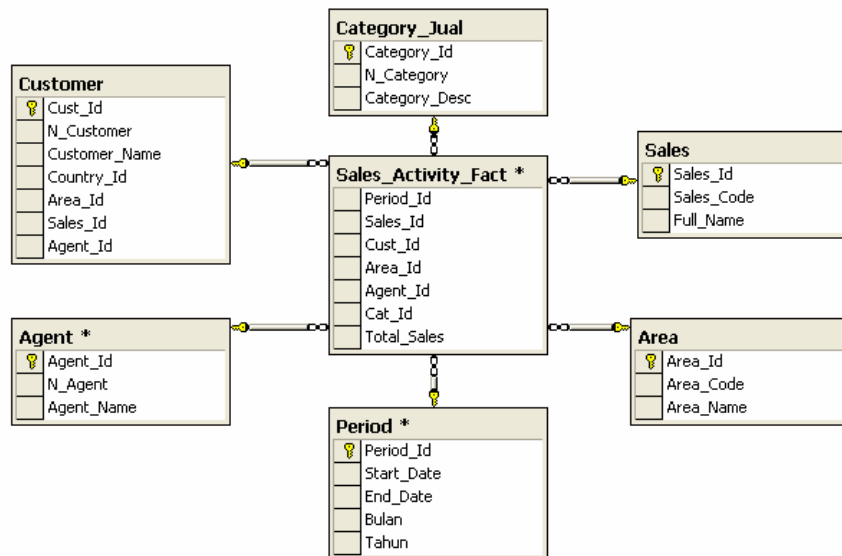


**Gambar 4.7 Skema Bintang Shipment\_Posting**

## 6. Skema Bintang Sales\_Activity

Skema bintang Sales\_Activity merupakan model dimensional penyajian informasi penjualan dari beberapa dimensi.

Skema bintang Sales\_Activity terdiri dari 1 tabel fakta dan 6 tabel dimensi. Tabel fakta yaitu Sales\_Activity\_Fact dan tabel dimensi yaitu Agent, Area, Category\_Jual, Customer, Period, dan Sales.

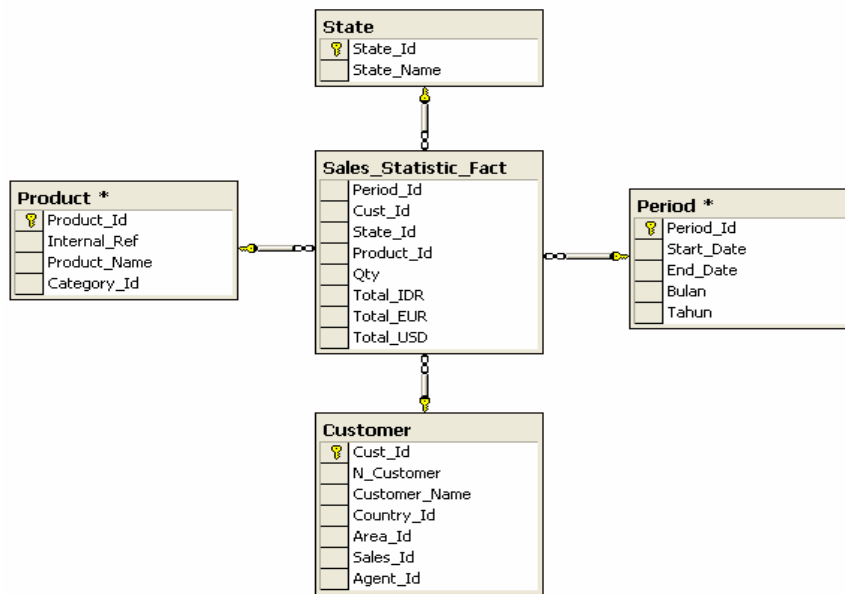


Gambar 4.8 Skema Bintang Sales\_Activity

## 7. Skema Bintang Sales\_Statistic

Skema bintang Sales\_Statistic merupakan model dimensional penyajian informasi statistik penjualan.

Skema bintang Sales\_Statistic terdiri dari 1 tabel fakta dan 4 tabel dimensi. Tabel fakta yaitu Sales\_Statistic\_Fact dan tabel dimensi yaitu Customer, Period, Product, dan State.



Gambar 4.9 Skema Bintang Sales\_Statistic

#### 4.3.4 Metadata

Metadata dari Independent Data Mart bagian *Inventory Control* dan *Sales PT Mane Indonesia* merupakan meta data administratif yang mencakup deskripsi basis data sumber, deskripsi tabel dimensi dan fakta. Deskripsi tabel dimensi dan fakta mencakup nama *field*, jenis dan ukuran data, proses transformasi data, sumber data, dan penjelasan singkat mengenai *field*.

Berikut ini adalah metadata administratif Independent Data Mart bagian *Inventory Control* dan *Sales PT Mane Indonesia*.

- **Tabel Stock\_Fact**

Identifikasi : Stock\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai stock yang tersedia di gudang

**Tabel 4.3 Stock\_Fact**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Period_Id	Int (4)	Transform	Period	Kode Waktu
Product_Id	Int (4)	Transform	Product	Kode Produk
Category_Id	Int (4)	Transform	Category	Kategori Produk
Available_Qty	Float (8)	Create	-	Jumlah Produk siap pakai

- **Tabel Internal\_Consumption\_Fact**

Identifikasi : Internal\_Consumption\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai konsumsi internal stock

**Tabel 4.4 Internal\_Consumption\_Fact**

<b>Id Field</b>	<b>Tipe &amp; Ukuran</b>	<b>Proses</b>	<b>Tabel Sumber</b>	<b>Deskripsi</b>
Period_Id	Int (4)	Transform	Period	Kode Waktu
Division_Id	Int (4)	Transform	Product	Kode Divisi
Product_Id	Int (4)	Transform	Category	Kode Produk
Motive_Id	Int (4)	Transform	Motive	Kode Jenis Transaksi
Qty	Float (8)	Create	-	Jumlah barang
Cost	Float (8)	Create	-	Harga barang

- **Tabel Shipment\_Post\_Fact**

Identifikasi : Shipment\_Post\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai pengiriman barang dengan invoice

**Tabel 4.5 Shipment\_Post\_Fact**

<b>Id Field</b>	<b>Tipe &amp; Ukuran</b>	<b>Proses</b>	<b>Tabel Sumber</b>	<b>Deskripsi</b>
Period_Id	Int (4)	Transform	Period	Kode Waktu
Product_Id	Int (4)	Transform	Product	Kode Produk
Category_Id	Int (4)	Transform	Category	Kategori Produk
Available_Qty	Float (8)	Create	-	Jumlah Produk siap pakai

- **Tabel Summarize\_Transaction\_Fact**

Identifikasi : Summarize\_Transaction\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai rangkuman transaksi stock

**Tabel 4.6 Summarize\_Transaction\_Fact**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Period_Id	Int (4)	Transform	Period	Kode Waktu
Motive_Id	Int (4)	Transform	Motive	Jenis transaksi
Qty	Float (8)	Create	-	Jumlah Produk
Cost	Float (8)	Create	-	Harga Produk

- **Tabel Sales\_Activity\_Fact**

Identifikasi : Sales\_Activity\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai penjualan berdasarkan *sales person*, pelanggan, wilayah, agen, dan kategori produk

**Tabel 4.7 Sales\_Activity\_Fact**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Period_Id	Int (4)	Transform	Period	Kode Waktu
Sales_Id	Int (4)	Transform	Sales	Kode Sales
Cust_Id	Int (4)	Transform	Category	Kode Pelanggan
Area_Id	Int (4)	Transform	Area	Jumlah Produk
Agent_Id	Int (4)	Transform	Agent	Harga Produk
Cat_Id	Int (4)	Transform	Category_Jual	Kategori Produk
Total_Sales	Float (8)	Create	-	Total Penjualan

- **Tabel Sales\_Statistic\_Fact**

Identifikasi : Sales\_Statistic\_Fact

Jenis : Tabel Fakta

Deskripsi : Data mengenai statistik penjualan

**Tabel 4.8 Sales\_Statistic\_Fact**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Period_Id	Int (4)	Transform	Period	Kode Waktu
Cust_Id	Int (4)	Transform	Motive	Kode Pelanggan

**Tabel 4.8 Lanjutan-Sales\_Statistic\_Fact**

Product_Id	Int (4)	Transform	Product	Kode Produk
State_Id	Int (4)	Transform	State	Kode Negara (Local/ Export)
Qty	Float (8)	Create	-	Jumlah Produk
Total_IDR	Float (8)	Create	-	Total Penjualan dalam Rupiah
Total_USD	Float (8)	Create	-	Total Penjualan dalam Dollar US
Total_EUR	Float (8)	Create	-	Total Penjualan dalam Euro

- **Tabel Agent**

Identifikasi : Agent

Jenis : Tabel Dimensi

Deskripsi : Data mengenai agen

**Tabel 4.9 Agent**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Agent_Id	Int (4)	Create	-	Kode Agent
N_Agent	Int (4)	Transform	Agent	Nomor Agent
Agent_Name	Varchar (60)	Transform	Agent	Nama Agent

- **Tabel Area**

Identifikasi : Area

Jenis : Tabel Dimensi

Deskripsi : Data mengenai wilayah penjualan

**Tabel 4.10 Agent**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Area_Id	Int (4)	Create	-	Kode Area
Area_Code	Varchar (2)	Transform	Client_Area	Nomor Area
Area_Name	nVarchar (2)	Transform	Client_Area	Nama Area

- **Tabel Category**

Identifikasi : Category

Jenis : Tabel Dimensi

Deskripsi : Data mengenai kategori internal produk

**Tabel 4.11 Category**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Cat_Id	Int (4)	Create	-	Kode Category
N_Cat	Smallint (2)	Transform	Glb_Category	Nomor Category
Cat_Desc	Varchar (25)	Transform	Glb_Category	Deskripsi Category

- **Tabel Category\_Jual**

Identifikasi : Category\_Jual

Jenis : Tabel Dimensi

Deskripsi : Data mengenai kategori produk yang dijual

**Tabel 4.12 Category\_Jual**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Category_Id	Int (4)	Create	-	Kode kategori jual
N_Category	Varchar (10)	Transform	External_List	Nomor kategori jual
Category_Desc	Varchar (50)	Transform	External_List	Deskripsi kategori jual

- **Tabel Country**

Identifikasi : Country

Jenis : Tabel Dimensi

Deskripsi : Data mengenai negara



**Tabel 4.13 Country**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Country_Id	Int (4)	Create	-	Nomor negara
N_Country	Int (4)	Transform	Glb_Country	Kode negara
Country_Name	Varchar (50)	Transform	Glb_Country	Nama negara

- **Tabel Currency**

Identifikasi : Currency

Jenis : Tabel Dimensi

Deskripsi : Data mengenai mata uang

**Tabel 4.14 Currency**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Currency_Id	Int (4)	Create	-	Kode mata uang
N_Currency	Varchar (3)	Transform	Currency	Nomor mata uang
Currency_Name	Varchar (50)	Transform	Currency	Nama mata uang

- **Tabel Customer**

Identifikasi : Customer

Jenis : Tabel Dimensi

Deskripsi : Data mengenai pelanggan

**Tabel 4.15 Customer**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Cust_Id	Int (4)	Create	-	Kode Pelanggan
N_Customer	Int (4)	Transform	Client	Nomor Pelanggan
Customer_Name	Varchar (80)	Transform	Client	Nama Pelanggan
Country_Id	Int (4)	Transform	Client	Kode Negara
Area_Id	Varchar (2)	Transform	Client_Address	Kode Wilayah

**Tabel 4.15 Lanjutan-Customer**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Sales_Id	Varchar (50)	Transform	Client	Kode Sales
Agent_Id	Int (4)	Transform	Client	Kode Agen

- **Tabel Division**

Identifikasi : Division

Jenis : Tabel Dimensi

Deskripsi : Data mengenai divisi pengguna stock

**Tabel 4.16 Division**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Division_Id	Int (4)	Create	-	Nomor divisi
Division_Code	nVarchar (3)	Create	-	Kode divisi
Division_Name	nVarchar (50)	Create	-	Nama divisi

- **Tabel Invoice**

Identifikasi : Invoice

Jenis : Tabel Dimensi

Deskripsi : Data mengenai invoice yang dikirimkan kepada pelanggan

**Tabel 4.17 Invoice**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Invoice_Id	Int (4)	Create	-	Indeks Invoice
Internal_No	Varchar (10)	Transform	Invoice_IC	Nomor Internal Invoice
Invoice_No	Varchar (20)	Transform	Invoice_IC	Nomor Invoice
Invoice_Date	Datetime (8)	Transform	Invoice_IC	Tanggal Invoice
Customer_Id	Int (4)	Transform	Invoice_IC	Kode Pelanggan
Currency_Id	Varchar (10)	Transform	Invoice_IC	Kode mata uang
Internal_Ref	Varchar (20)	Transform	Invoice_IC	Kode internal barang

**Tabel 4.17 Lanjutan-Invoice**

External_Ref	Varchar (20)	Transform	Invoice_IC	Kode eksternal barang
Quantity	Money (8)	Transform	Invoice_IC	Jumlah barang
Price	Money (8)	Transform	Invoice_IC	Harga barang

- **Tabel Motive**

Identifikasi : Motive

Jenis : Tabel Dimensi

Deskripsi : Data mengenai jenis transaksi stock

**Tabel 4.18 Motive**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Motive_Id	Int (4)	Create	-	Indeks jenis transaksi
N_Motive	Varchar (1)	Transform	Motive	Kode jenis transaksi
In_Out	Varchar (1)	Transform	Motive	Transaksi Masuk/ Keluar
Motive_Desc	Char (20)	Transform	Motive	Deskripsi transaksi

- **Tabel Period**

Identifikasi : Period

Jenis : Tabel Dimensi

Deskripsi : Data mengenai waktu

**Tabel 4.19 Period**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Period_Id	Int (4)	Create	-	Indeks periode
Start_Date	Datetime (8)	Transform	PostingPeriodCS	Tanggal awal periode posting

**Tabel 4.19 Lanjutan-Period**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
End_Date	Datetime (8)	Transform	PostingPeriodCS	Tanggal akhir periode posting
Bulan	Int (4)	Transform	PostingPeriodCS	Bulan periode posting
Tahun	Int (4)	Transform	PostingPeriodCS	Tahun periode posting

- **Tabel Product**

Identifikasi : Product

Jenis : Tabel Dimensi

Deskripsi : Data mengenai produk

**Tabel 4.20 Product**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Product_Id	Int (4)	Create	-	Indeks produk
Internal_Ref	Varchar (11)	Transform	Product_Identification	Kode internal barang
Product_Name	Varchar (60)	Transform	Product_Identification	Nama barang
Category_Id	nVarchar (2)	Transform	Product_Identification	Kategori barang
Division	Varchar (3)	Transform	Product_Identification	Divisi

- **Tabel Sales**

Identifikasi : Sales

Jenis : Tabel Dimensi

Deskripsi : Data mengenai *sales person*

**Tabel 4.21 Sales**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
Sales_Id	Int (4)	Create	-	Indeks Sales
Sales_Code	Varchar (50)	Transform	Sec_User	Kode Sales
Full_Name	nVarchar (50)	Transform	Sec_User	Nama lengkap sales

- **Tabel State**

Identifikasi : State

Jenis : Tabel Dimensi

Deskripsi : Data mengenai bagian penjualan

**Tabel 4.22 State**

Id Field	Tipe & Ukuran	Proses	Tabel Sumber	Deskripsi
State_Id	Int (4)	Create	-	Kode bagian
State_Name	Varchar (15)	Create	-	Nama bagian (ekspor/lokal)

#### 4.4 Proses ETL (*Extraction, Transformation, Loading*)

Proses ETL (*Extraction, Transformasion, Loading*) merupakan sebuah proses yang memungkinkan pemindahan data dari beberapa basis data operasional, melakukan *reformat* dan membersihkan data, dan kemudian mengisinya ke dalam sebuah basis data analisis.

##### 4.4.1 Extract

Pada tahapan *extract* dilakukan pemilihan data dari beberapa basis data operasional yang berhubungan dengan bagian *Inventory Control* dan *Sales*.

Data yang sudah dipilih dari beberapa basis data operasional disimpan ke dalam sebuah basis data rekonsiliasi yang bertujuan untuk pemusatan data sebelum ditransformasi dan dikirimkan ke basis data analisis yaitu *independent data mart*.

Berikut ini merupakan langkah pengambilan data dari beberapa basis data operasional:

```
use [MANEBuffer]
GO

CREATE VIEW dbo.Agent
AS
SELECT N_Agent, AgentName, City, Country, Contact
FROM SysSQLClone.dbo.Agent

CREATE VIEW dbo.Category
AS
SELECT * FROM PbookSQL.dbo.Category

CREATE VIEW dbo.Client
AS
SELECT * FROM SysSQLClone.dbo.Client

CREATE VIEW dbo.Client_Area
AS
SELECT N_Client, Area
FROM SysSQLNet.dbo.Client_Address
WHERE (Delivery = 1)
GO
```

```

CREATE VIEW dbo.Currency
AS
SELECT * FROM PbookSQL.dbo.Currency
GO

CREATE VIEW dbo.DO
AS
SELECT * FROM EXIS.dbo.DO
GO

CREATE VIEW dbo.DO_Collie_Contents
AS
SELECT * FROM EXIS.dbo.DO_Collie_Contents
GO

CREATE VIEW dbo.DO_Posted
AS
SELECT * FROM EXIS.dbo.DO_Posted
GO

CREATE VIEW dbo.External_List
AS
SELECT Customer_Ref, Internal_Ref, External_Ref, External_Name, Label,
Category, Product_Counter, Prod_site, active, MUI_Process, RM, Sales_Person,
Segment_Code, FDA_No, COA_Yes, COA_No, AdditionalInfo1, AdditionalInfo2
FROM PbookSQL.dbo.External_List
GO

CREATE VIEW dbo.Glb_Area
AS
SELECT SysSQLNet.dbo.Client_Area.*
FROM SysSQLNet.dbo.Client_Area
GO

CREATE VIEW dbo.Glb_Category
AS
SELECT * FROM SysSQLClone.dbo.Glb_Category
GO

CREATE VIEW dbo.Glb_Country
AS
SELECT * FROM SysSQLClone.dbo.Glb_Country
GO

CREATE VIEW dbo.Internal_Consumption_Buffer
AS
SELECT MANEMART.dbo.Period.Period_Id, MANEMART.dbo.Division.Division_Id AS
Div_Id, MANEMART.dbo.Product.Product_Id,
MANEMART.dbo.Motive.Motive_Id, dbo.Stock_In_Out_Buffer.Qty,
dbo.Stock_In_Out_Buffer.Cost_Avg
FROM dbo.Stock_In_Out_Buffer INNER JOIN
MANEMART.dbo.Period ON dbo.Stock_In_Out_Buffer.[Month] =
MANEMART.dbo.Period.Bulan AND dbo.Stock_In_Out_Buffer.[Year] =
MANEMART.dbo.Period.Tahun INNER JOIN
MANEMART.dbo.Product ON dbo.Stock_In_Out_Buffer.Internal_Ref =
MANEMART.dbo.Product.Internal_Ref INNER JOIN
MANEMART.dbo.Motive ON dbo.Stock_In_Out_Buffer.Motive =
MANEMART.dbo.Motive.N_Motive INNER JOIN
MANEMART.dbo.Division ON MANEMART.dbo.Product.Division =

```

```
MANEMART.dbo.Division.Division_Code
GO

CREATE VIEW dbo.Invoice_Buffer
AS
SELECT dbo.[Sales Invoice Header].Internal_No, dbo.[Sales Invoice
Header].Document_No, dbo.[Sales Invoice Header].[Date],
dbo.[Sales Invoice Header].Customer_Ref, dbo.[Sales Invoice
Header].Currency, dbo.[Sales Invoice Detail].Internal_Ref,
dbo.[Sales Invoice Detail].External_Ref, dbo.[Sales Invoice
Detail].Quantity, dbo.[Sales Invoice Detail].Price
FROM dbo.[Sales Invoice Header] INNER JOIN
dbo.[Sales Invoice Detail] ON dbo.[Sales Invoice Header].Internal_No =
dbo.[Sales Invoice Detail].Internal_No
GO

CREATE VIEW dbo.Invoice_Detail_IC
AS
SELECT IC.dbo.Invoice_Detail_IC.*
FROM IC.dbo.Invoice_Detail_IC
GO

CREATE VIEW dbo.Invoice_Detail_Posted
AS
SELECT AR_SQL.dbo.Invoice_Detail_Posted.*
FROM AR_SQL.dbo.Invoice_Detail_Posted
GO

CREATE VIEW dbo.Invoice_IC
AS
SELECT IC.dbo.Invoice_IC.*
FROM IC.dbo.Invoice_IC
GO

CREATE VIEW dbo.Invoice_IC_Buffer
AS
SELECT      dbo.Invoice_IC.Int_No, dbo.Invoice_IC.Inv_No,
dbo.Invoice_IC.Inv_Date, dbo.Invoice_IC.Customer_Ref,
dbo.Invoice_Detail_IC.Internal_Ref,
dbo.Invoice_Detail_IC.External_Ref, dbo.Invoice_Detail_IC.Category,
dbo.Invoice_Detail_IC.Dos_No, dbo.Invoice_Detail_IC.Lot_No,
dbo.Invoice_Detail_IC.Qty, dbo.Invoice_Detail_IC.SellingPrice
FROM  dbo.Invoice_IC INNER JOIN
dbo.Invoice_Detail_IC ON dbo.Invoice_IC.Int_No =
dbo.Invoice_Detail_IC.Int_No
GO

CREATE VIEW dbo.Invoice_Posted
AS
SELECT AR_SQL.dbo.Invoice_Posted.*
FROM  AR_SQL.dbo.Invoice_Posted
GO
GO

CREATE VIEW dbo.Lot_Transaction
AS
SELECT  EXIS.dbo.Lot_Transaction.*
FROM EXIS.dbo.Lot_Transaction
GO
```



```

CREATE VIEW dbo.Market_Segment
AS
SELECT * FROM PbookSQL.dbo.Market_Segment
GO

CREATE VIEW dbo.MonthlyPostingCS
AS
SELECT * FROM PSIS_CS.dbo.MonthlyPostingCS
GO

CREATE VIEW dbo.Montly_Rekap
AS
SELECT AR_SQL.dbo.Montly_Rekap.* FROM AR_SQL.dbo.Montly_Rekap
GO

CREATE VIEW dbo.Motive
AS
SELECT FROM PSIS_SQL.dbo.Motive
GO

CREATE VIEW dbo.PostingPeriodCS
AS
SELECT * FROM PSIS_CS.dbo.PostingPeriodCS
GO

CREATE VIEW dbo.Product_Identification
AS
SELECT *
FROM SysSQLClone.dbo.Product_Identification
GO

CREATE VIEW dbo.Product_Stock
AS
SELECT * FROM PSIS_SQL.dbo.Product_Stock
GO

CREATE VIEW dbo.[Sales Invoice Detail]
AS
SELECT NavisionBuffer.dbo.[Sales Invoice Detail].*
FROM NavisionBuffer.dbo.[Sales Invoice Detail]
GO

CREATE VIEW dbo.[Sales Invoice Header]
AS
SELECT NavisionBuffer.dbo.[Sales Invoice Header].*
FROM NavisionBuffer.dbo.[Sales Invoice Header]
GO

CREATE VIEW dbo.Sales_Buffer
AS
SELECT MANEMART.dbo.Period.Period_Id, MANEMART.dbo.Sales.Sales_Id,
MANEMART.dbo.Customer.Cust_Id, MANEMART.dbo.Area.Area_Id,
MANEMART.dbo.Agent.Agent_Id, MANEMART.dbo.Product.Product_Id,
MANEMART.dbo.Category_Jual.Category_Id,
dbo.Montly_Rekap.Quantity_IO AS Qty, dbo.Montly_Rekap.Total AS Total_IDR,

```

```

dbo.Montly_Rekap.TotalFRF AS Total_EUR ,
dbo.Montly_Rekap.TotalUSD AS Total_USD
FROM MANEMART.dbo.Agent RIGHT OUTER JOIN
MANEMART.dbo.Sales RIGHT OUTER JOIN
MANEMART.dbo.Customer ON MANEMART.dbo.Sales.Sales_Code =
MANEMART.dbo.Customer.Sales_Id LEFT OUTER JOIN
MANEMART.dbo.Area ON MANEMART.dbo.Customer.Area_Id =
MANEMART.dbo.Area.Area_Code ON
MANEMART.dbo.Agent.Agent_Id = MANEMART.dbo.Customer.Agent_Id RIGHT OUTER
JOIN
MANEMART.dbo.Product INNER JOIN
dbo.Montly_Rekap ON MANEMART.dbo.Product.Internal_Ref =
dbo.Montly_Rekap.Internal_Ref ON
MANEMART.dbo.Customer.N_Customer = dbo.Montly_Rekap.Customer_Ref LEFT OUTER
JOIN
MANEMART.dbo.Period ON dbo.Montly_Rekap.Period = MANEMART.dbo.Period.Bulan
AND
dbo.Montly_Rekap.[Year] = MANEMART.dbo.Period.Tahun LEFT OUTER JOIN
MANEMART.dbo.Category_Jual RIGHT OUTER JOIN
dbo.External_List ON MANEMART.dbo.Category_Jual.N_Category =
dbo.External_List.Category ON
dbo.Montly_Rekap.Customer_Ref = dbo.External_List.Customer_Ref AND
dbo.Montly_Rekap.Internal_Ref = dbo.External_List.Internal_Ref AND
dbo.Montly_Rekap.External_Ref = dbo.External_List.External_Ref

GO

CREATE VIEW dbo.Sales_Invoice
AS
SELECT      dbo.[Sales Invoice Header].Internal_No, dbo.[Sales Invoice
Header].Document_No, dbo.[Sales Invoice Header].Type, dbo.[Sales Invoice
Header].[Date],
MONTH(dbo.[Sales Invoice Header].[Date]) AS Period, YEAR(dbo.[Sales Invoice
Header].[Date]) AS Year,
CAST(RIGHT(dbo.[Sales Invoice Header].Customer_Ref, 6) AS int) AS
Customer_Ref, dbo.[Sales Invoice Header].Currency,
dbo.[Sales Invoice Header].Tax, dbo.[Sales Invoice Header].Freight,
dbo.[Sales Invoice Header].Discount_Amount, dbo.[Sales Invoice Header].Rate,
dbo.[Sales Invoice Header].Include_Tax, dbo.[Sales Invoice Header].USD_Rate,
dbo.[Sales Invoice Header].EUR_Rate,
dbo.[Sales Invoice Header].Price_Correction, dbo.[Sales Invoice
Detail].DOS_No, dbo.[Sales Invoice Detail].DOS_Ref,
dbo.[Sales Invoice Detail].Internal_Ref, dbo.[Sales Invoice
Detail].External_Ref, SUM(dbo.[Sales Invoice Detail].Quantity) AS Quantity,
SUM(dbo.[Sales Invoice Detail].Quantity * dbo.[Sales Invoice Detail].Price)
/ SUM(dbo.[Sales Invoice Detail].Quantity) AS AvgSellingPrice,
SUM(dbo.[Sales Invoice Detail].Line_Discount_Amount) / SUM(dbo.[Sales
Invoice Detail].Quantity) AS AvgLineDiscountAmount,
SUM(dbo.[Sales Invoice Detail].Total_IDR) AS Total_IDR, SUM(dbo.[Sales
Invoice Detail].Total_EUR) AS Total_EUR,
SUM(dbo.[Sales Invoice Detail].Total_USD) AS Total_USD
FROM dbo.[Sales Invoice Detail] INNER JOIN
dbo.[Sales Invoice Header] ON dbo.[Sales Invoice Detail].Internal_No =
dbo.[Sales Invoice Header].Internal_No
GROUP BY dbo.[Sales Invoice Header].Internal_No, dbo.[Sales Invoice
Header].Document_No, dbo.[Sales Invoice Header].Type,
dbo.[Sales Invoice Header].[Date], CAST(RIGHT(dbo.[Sales Invoice
Header].Customer_Ref, 6) AS int), dbo.[Sales Invoice Header].Currency,
dbo.[Sales Invoice Header].Tax, dbo.[Sales Invoice Header].Freight,
dbo.[Sales Invoice Header].Discount_Amount, dbo.[Sales Invoice Header].Rate,

```

```

dbo.[Sales Invoice Header].Include_Tax, dbo.[Sales Invoice Header].USD_Rate,
dbo.[Sales Invoice Header].EUR_Rate,
dbo.[Sales Invoice Detail].DOS_No, dbo.[Sales Invoice Detail].Internal_Ref,
dbo.[Sales Invoice Detail].External_Ref,
dbo.[Sales Invoice Detail].DOS_Ref, dbo.[Sales Invoice
Header].Price_Correction, MONTH(dbo.[Sales Invoice Header].[Date]),
YEAR(dbo.[Sales Invoice Header].[Date])
HAVING (dbo.[Sales Invoice Detail].Internal_Ref > '')

CREATE VIEW dbo.Sales_Invoice_Buffer
AS
SELECT      dbo.Sales_Invoice.Internal_No, dbo.Sales_Invoice.Document_No,
dbo.Sales_Invoice.[Date], dbo.Sales_Invoice.Customer_Ref,
dbo.Sales_Invoice.Currency, MANEMART.dbo.Product.Product_Id,
dbo.Sales_Invoice.External_Ref, dbo.Sales_Invoice.Quantity,
dbo.Sales_Invoice.Total_IDR, dbo.Sales_Invoice.Total_EUR,
dbo.Sales_Invoice.Total_USD
FROM      dbo.Sales_Invoice INNER JOIN
MANEMART.dbo.Product ON dbo.Sales_Invoice.Internal_Ref =
MANEMART.dbo.Product.Internal_Ref LEFT OUTER JOIN
MANEMART.dbo.Period ON dbo.Sales_Invoice.Period = MANEMART.dbo.Period.Bulan
AND
dbo.Sales_Invoice.[Year] = MANEMART.dbo.Period.Tahun LEFT OUTER JOIN
MANEMART.dbo.Currency ON dbo.Sales_Invoice.Currency =
MANEMART.dbo.Currency.Currency_Id LEFT OUTER JOIN
MANEMART.dbo.Customer ON dbo.Sales_Invoice.Customer_Ref =
MANEMART.dbo.Customer.N_Customer

GO

CREATE VIEW dbo.Sales_Person
AS
SELECT UserID, FirstName + N' ' + LastName AS Full_Name
FROM      SysSQLClone.dbo.SecUser
WHERE (Department = 'sld')
GO

CREATE VIEW dbo.Shipment_Post_Buffer
AS
SELECT      MANEMART.dbo.Period.Period_Id, MANEMART.dbo.Invoice.Invoice_Id,
MANEMART.dbo.Customer.Cust_Id, MANEMART.dbo.Product.Product_Id,
MANEMART.dbo.Category_Jual.Category_Id, dbo.Shipment_Post_Buffer1.Qty,
dbo.Shipment_Post_Buffer1.SellingPrice AS Cost,
dbo.Shipment_Post_Buffer1.Cost_Avg AS COGS
FROM      MANEMART.dbo.Category_Jual INNER JOIN
dbo.Shipment_Post_Buffer1 INNER JOIN
MANEMART.dbo.Period ON dbo.Shipment_Post_Buffer1.[Month] =
MANEMART.dbo.Period.Bulan AND
dbo.Shipment_Post_Buffer1.[Year] = MANEMART.dbo.Period.Tahun INNER JOIN
MANEMART.dbo.Invoice ON dbo.Shipment_Post_Buffer1.Inv_No =
MANEMART.dbo.Invoice.Invoice_No INNER JOIN
MANEMART.dbo.Customer ON dbo.Shipment_Post_Buffer1.Customer_Ref =
MANEMART.dbo.Customer.N_Customer INNER JOIN
MANEMART.dbo.Product ON dbo.Shipment_Post_Buffer1.Internal_Ref =
MANEMART.dbo.Product.Internal_Ref ON
MANEMART.dbo.Category_Jual.N_Category = dbo.Shipment_Post_Buffer1.Category

GO

```

```

CREATE VIEW dbo.Stock_Buffer
AS
SELECT      MANEMART.dbo.Period.Period_Id, MANEMART.dbo.Product.Product_Id,
MANEMART.dbo.Category.Cat_Id, dbo.Stock_for_Buffer.Available_Qty,
dbo.Stock_for_Buffer.Cost
FROM      dbo.Stock_for_Buffer INNER JOIN
MANEMART.dbo.Period ON dbo.Stock_for_Buffer.[Month] =
MANEMART.dbo.Period.Bulan AND
dbo.Stock_for_Buffer.[Year] = MANEMART.dbo.Period.Tahun INNER JOIN
MANEMART.dbo.Product ON dbo.Stock_for_Buffer.Internal_Ref =
MANEMART.dbo.Product.Internal_Ref INNER JOIN
MANEMART.dbo.Category ON MANEMART.dbo.Product.Category_Id =
MANEMART.dbo.Category.N_Cat

GO

CREATE VIEW dbo.Stock_DOPosted
AS
SELECT      dbo.Stock_In_Out.Internal_Ref, dbo.Stock_In_Out.[Date],
dbo.Stock_In_Out.Qty, dbo.DO_Posted.DOS_No, dbo.DO_Posted.Lot_No,
dbo.Stock_In_Out.Cost_Avg
FROM      dbo.DO_Posted INNER JOIN
dbo.Stock_In_Out ON dbo.DO_Posted.Internal_Ref =
dbo.Stock_In_Out.Internal_Ref AND dbo.DO_Posted.[Date] =
dbo.Stock_In_Out.[Date] AND
dbo.DO_Posted.In_Out = dbo.Stock_In_Out.In_Out
WHERE      (dbo.Stock_In_Out.In_Out = 'O') AND (dbo.Stock_In_Out.Motive = 'G')

GO

CREATE VIEW dbo.Stock_In_Out
AS
SELECT      * FROM PSIS_SQL.dbo.Stock_In_Out
GO

CREATE VIEW dbo.Stock_Buffer
AS
SELECT      dbo.Stock_In_Out_Buffer.[Month], dbo.Stock_In_Out_Buffer.[Year],
dbo.Stock_In_Out_Buffer.Internal_Ref, dbo.Product_Stock.Available_Qty,
dbo.Product_Stock.Cost_Avg AS Cost
FROM      dbo.Stock_In_Out_Buffer INNER JOIN
dbo.Product_Stock ON
dbo.Stock_In_Out_Buffer.Internal_Ref = dbo.Product_Stock.Internal_Ref
GO

CREATE VIEW dbo.Summarize_Transaction_Buffer
AS
SELECT      MANEMART.dbo.Period.Period_Id, MANEMART.dbo.Motive.Motive_Id,
dbo.Stock_In_Out_Buffer.Qty, dbo.Stock_In_Out_Buffer.Cost_Avg
FROM      dbo.Stock_In_Out_Buffer INNER JOIN
MANEMART.dbo.Period ON dbo.Stock_In_Out_Buffer.[Month]
= MANEMART.dbo.Period.Bulan AND
dbo.Stock_In_Out_Buffer.[Year] =
MANEMART.dbo.Period.Tahun INNER JOIN
MANEMART.dbo.Motive ON dbo.Stock_In_Out_Buffer.Motive
= MANEMART.dbo.Motive.N_Motive
GO

```

#### 4.4.2 Transformasi

Pada tahapan transformasi dilakukan *reformat* data dan pemilihan data-data yang diperlukan untuk independent data mart. Tahap ini menghasilkan 14 buah tabel dimensi dan 6 buah tabel fakta. Berikut ini merupakan langkah-langkah transformasi data:

```

use [MANEMART]
GO

CREATE TABLE [dbo].[Agent] (
    [Agent_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Agent] [int] NULL ,
    [Agent_Name] [varchar] (60) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Area] (
    [Area_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Area_Code] [varchar] (2) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
    [Area_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Category] (
    [Cat_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Cat] [smallint] NULL ,
    [Cat_Desc] [varchar] (25) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Category_Jual] (
    [Category_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Category] [varchar] (10) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL ,
    [Category_Desc] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Country] (
    [Country_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Country] [int] NOT NULL ,
    [Country_Name] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Currency] (
    [Currency_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Currency] [varchar] (3) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
,
    [Currency_Name] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]

```

```

GO

CREATE TABLE [dbo].[Customer] (
    [Cust_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Customer] [int] NOT NULL ,
    [Customer_Name] [varchar] (80) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Country_Id] [int] NULL ,
    [Area_Id] [varchar] (2) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
    [Sales_Id] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
    [Agent_Id] [int] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Division] (
    [Division_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Division_Code] [nvarchar] (3) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Division_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Internal_Consumption_Fact] (
    [Period_Id] [int] NULL ,
    [Div_Id] [int] NULL ,
    [Product_Id] [int] NULL ,
    [Motive_Id] [int] NULL ,
    [Qty] [float] NULL ,
    [Cost] [float] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Invoice] (
    [Invoice_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Internal_No] [varchar] (10) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Invoice_No] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
,
    [Invoice_Date] [datetime] NULL ,
    [Customer_Id] [int] NULL ,
    [Currency_Id] [varchar] (10) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Internal_Ref] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [External_Ref] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Quantity] [money] NULL ,
    [Price] [money] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Motive] (
    [Motive_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [N_Motive] [varchar] (1) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL ,
    [In_Out] [varchar] (1) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
    [Motive_Desc] [char] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

```

```

CREATE TABLE [dbo].[Period] (
    [Period_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Start_Date] [datetime] NULL ,
    [End_Date] [datetime] NULL ,
    [Bulan] [int] NULL ,
    [Tahun] [int] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Product] (
    [Product_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Internal_Ref] [varchar] (11) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Product_Name] [varchar] (60) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Category_Id] [nvarchar] (2) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
    [Division] [varchar] (3) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Sales] (
    [Sales_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [Sales_Code] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
,
    [Full_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Sales_Activity_Fact] (
    [Period_Id] [int] NULL ,
    [Sales_Id] [int] NULL ,
    [Cust_Id] [int] NULL ,
    [Area_Id] [int] NULL ,
    [Agent_Id] [int] NULL ,
    [Category_Id] [int] NULL ,
    [Total_Sales] [float] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Sales_Statistic_Fact] (
    [Period_Id] [int] NULL ,
    [Cust_Id] [int] NULL ,
    [State_Id] [int] NULL ,
    [Product_Id] [int] NULL ,
    [Qty] [float] NULL ,
    [Total_IDR] [float] NULL ,
    [Total_EUR] [float] NULL ,
    [Total_USD] [float] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Shipment_Post_Fact] (
    [Period_Id] [int] NULL ,
    [Invoice_Id] [int] NULL ,
    [Customer_Id] [int] NULL ,
    [Currency_Id] [int] NULL ,
    [Product_Id] [int] NULL ,
    [Category_Id] [int] NULL ,

```

```
        [Qty] [float] NULL ,
        [Cost] [float] NULL ,
        [COGS] [float] NULL
    ) ON [PRIMARY]
GO

CREATE TABLE [dbo].[State] (
    [State_Id] [int] IDENTITY (1, 1) NOT NULL ,
    [State_Name] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Stock_Fact] (
    [Period_Id] [int] NULL ,
    [Product_Id] [int] NULL ,
    [Category_Id] [int] NULL ,
    [Available_Qty] [float] NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Summarize_Transaction_Fact] (
    [Period_Id] [int] NULL ,
    [Motive_Id] [int] NULL ,
    [Qty] [float] NULL ,
    [Cost] [float] NULL
) ON [PRIMARY]
GO
```



Berikut ini adalah struktur *data store* pada *independent data mart* setelah proses transformasi:



Gambar 4.10 Hasil Transformasi

### 4.4.3 Loading

Tahapan *loading* dilakukan setelah tahapan transformasi selesai dilakukan. Pada tahap loading dilakukan pengisian data ke *independent data*. Berikut ini merupakan langkah-langkah *loading* data ke *independent data mart*:

```

if exists (select * from dbo.sysobjects where id =
object_id(N'[dbo].[AddDimensionData]') and OBJECTPROPERTY(id,
N'IsProcedure') = 1)
drop procedure [dbo].[AddDimensionData]
GO

if exists (select * from dbo.sysobjects where id =
object_id(N'[dbo].[AddFactData]') and OBJECTPROPERTY(id, N'IsProcedure') =
1)
drop procedure [dbo].[AddFactData]
GO

CREATE PROCEDURE dbo.AddDimensionData AS

/* INSERT Agent */
INSERT INTO MANEMART.dbo.Agent(N_Agent,Agent_Name)
SELECT Agent.N_Agent, Agent.AgentName
FROM Agent LEFT OUTER JOIN
MANEMART.dbo.Agent ON Agent.N_Agent = MANEMART.dbo.Agent.N_Agent
WHERE (MANEMART.dbo.Agent.N_Agent IS NULL)

/* INSERT Area */
INSERT INTO MANEMART.dbo.Area (Area_Code, Area_Name)
SELECT Glb_Area.Code_Area, Glb_Area.Area_Name
FROM Glb_Area LEFT OUTER JOIN
MANEMART.dbo.Area ON Glb_Area.Code_Area = MANEMART.dbo.Area.Area_Code
WHERE (MANEMART.dbo.Area.Area_Code IS NULL)

/* INSERT Category*/
INSERT INTO MANEMART.dbo.Category (N_Cat, Cat_Desc)
SELECT dbo.Glb_Category.[No], dbo.Glb_Category.Description
FROM MANEMART.dbo.Category RIGHT OUTER JOIN
dbo.Glb_Category ON MANEMART.dbo.Category.N_Cat = dbo.Glb_Category.[No]
WHERE (MANEMART.dbo.Category.N_Cat IS NULL)

/* INSERT Category_Jual */
INSERT INTO MANEMART.dbo.Category_Jual (N_Category,Category_Desc)
SELECT Category.Category, Category.Description
FROM Category LEFT OUTER JOIN
MANEMART.dbo.Category_Jual ON Category.Category =
MANEMART.dbo.Category_Jual.N_Category
WHERE (MANEMART.dbo.Category_Jual.N_Category IS NULL)

/* INSERT Currency */
INSERT INTO MANEMART.dbo.Currency (N_Currency,Currency_Name)
SELECT Currency_Ref,Description
FROM Currency LEFT OUTER JOIN
MANEMART.dbo.Currency ON Currency.Currency_Ref =
MANEMART.dbo.Currency.N_Currency

```

```

WHERE (MANEMART.dbo.Currency.N_Currency IS NULL)

/*INSERT Country */
INSERT INTO MANEMART.dbo.Country (N_Country, Country_Name)
SELECT Glb_Country.N_Country, Glb_Country.CountryName
FROM Glb_Country LEFT OUTER JOIN
MANEMART.dbo.Country ON Glb_Country.N_Country =
MANEMART.dbo.Country.N_Country
WHERE (MANEMART.dbo.Country.N_Country IS NULL)

/* INSERT INTO CUSTOMER */
INSERT INTO MANEMART.dbo.Customer (N_Customer, Customer_Name, Country_Id,
Area_Id, Sales_Id, Agent_Id)
SELECT Client.N_Client, Client.ClientName,
Client.Country, Client_Area.Area, Client.Sales_P1, Client.N_Agent
FROM Client LEFT OUTER JOIN
Client_Area ON Client.N_Client = Client_Area.N_Client LEFT OUTER JOIN
MANEMART.dbo.Customer ON Client.N_Client = MANEMART.dbo.Customer.N_Customer
WHERE (MANEMART.dbo.Customer.N_Customer IS NULL)

/* Invoice */
INSERT INTO MANEMART.dbo.Invoice
(Internal_No, Invoice_No, Invoice_Date, Customer_Id, Currency_Id,
Internal_Ref, External_Ref, Quantity, Price)

SELECT Invoice_Buffer.Internal_No, Invoice_Buffer.Document_No,
Invoice_Buffer.[Date], CAST(RIGHT(Invoice_Buffer.Customer_Ref, 6) AS int) AS
Cust_Ref, Invoice_Buffer.Currency, Invoice_Buffer.Internal_Ref,
Invoice_Buffer.External_Ref, Invoice_Buffer.Quantity, Invoice_Buffer.Price
FROM Invoice_Buffer LEFT OUTER JOIN
MANEMART.dbo.Invoice ON Invoice_Buffer.Internal_No =
MANEMART.dbo.Invoice.Internal_No AND
Invoice_Buffer.Document_No = MANEMART.dbo.Invoice.Invoice_No
WHERE (MANEMART.dbo.Invoice.Internal_No IS NULL) AND
(MANEMART.dbo.Invoice.Invoice_No IS NULL)

/* INSERT MOTIVE */
INSERT INTO MANEMART.dbo.Motive (N_Motive, In_Out, Motive_Desc)
SELECT Motive.ID, Motive.In_Out_Status, Motive.Motive
FROM Motive LEFT OUTER JOIN
MANEMART.dbo.Motive ON Motive.Motive = MANEMART.dbo.Motive.N_Motive
WHERE (MANEMART.dbo.Motive.N_Motive IS NULL)

/* INSERT PERIOD */
INSERT INTO MANEMART.dbo.Period (Start_Date, End_Date, Bulan, Tahun)
SELECT PostingPeriodCS.StartDate, PostingPeriodCS.EndDate,
PostingPeriodCS.Period, PostingPeriodCS.[Year]
FROM PostingPeriodCS LEFT OUTER JOIN
MANEMART.dbo.Period ON PostingPeriodCS.[Year] = MANEMART.dbo.Period.Tahun
AND PostingPeriodCS.Period = MANEMART.dbo.Period.Bulan
WHERE (MANEMART.dbo.Period.Bulan IS NULL) AND (MANEMART.dbo.Period.Tahun IS
NULL)

/*INSERT INTO Product */
INSERT INTO MANEMART.dbo.Product (Internal_Ref, Product_Name, Category_Id,
Division)
SELECT Product_Identification.Reference, Product_Identification.ProductName,
Product_Identification.Category, Product_Identification.Responsible
FROM Product_Identification LEFT OUTER JOIN
MANEMART.dbo.Product ON Product_Identification.Reference =

```

```

MANEMART.dbo.Product.Internal_Ref
WHERE (MANEMART.dbo.Product.Internal_Ref IS NULL)

/* SALES */
INSERT INTO MANEMART.dbo.Sales (Sales_Code,Full_Name)
SELECT Sales_Person.UserID, Sales_Person.Full_Name
FROM Sales_Person LEFT OUTER JOIN
MANEMART.dbo.Sales ON Sales_Person.UserID = MANEMART.dbo.Sales.Sales_Code
WHERE (MANEMART.dbo.Sales.Sales_Code IS NULL)

/* STATE */
GO
SET QUOTED_IDENTIFIER OFF
GO
SET ANSI_NULLS ON
GO

CREATE PROCEDURE dbo.AddFactData AS

/* INSERT INTO SUMMARIZE_TRANSACTION_FACT */

INSERT INTO MANEMART.dbo.Summarize_Transaction_Fact
(Period_Id,Motive_Id,Qty,Cost)

SELECT dbo.Summarize_Transaction_Buffer.Period_Id,
dbo.Summarize_Transaction_Buffer.Motive_Id,
SUM(dbo.Summarize_Transaction_Buffer.Qty) AS Qty,
SUM(dbo.Summarize_Transaction_Buffer.Qty *
dbo.Summarize_Transaction_Buffer.Cost_Avg) AS Cost
FROM dbo.Summarize_Transaction_Buffer LEFT OUTER JOIN
MANEMART.dbo.Summarize_Transaction_Fact ON
dbo.Summarize_Transaction_Buffer.Period_Id =
MANEMART.dbo.Summarize_Transaction_Fact.Period_Id AND
dbo.Summarize_Transaction_Buffer.Motive_Id =
MANEMART.dbo.Summarize_Transaction_Fact.Motive_Id
WHERE (MANEMART.dbo.Summarize_Transaction_Fact.Motive_Id IS NULL) AND
(MANEMART.dbo.Summarize_Transaction_Fact.Period_Id IS NULL)
GROUP BY dbo.Summarize_Transaction_Buffer.Period_Id,
dbo.Summarize_Transaction_Buffer.Motive_Id

/*INSERT INTO STOCK_FACT */
INSERT INTO MANEMART.dbo.Stock_Fact
(Period_Id,Product_Id,Category_Id,Available_Qty)
SELECT Stock_Buffer.Period_Id, Stock_Buffer.Product_Id, Stock_Buffer.Cat_Id,
Stock_Buffer.Available_Qty
FROM Stock_Buffer LEFT OUTER JOIN
MANEMART.dbo.Stock_Fact ON Stock_Buffer.Period_Id =
MANEMART.dbo.Stock_Fact.Period_Id AND
Stock_Buffer.Product_Id = MANEMART.dbo.Stock_Fact.Product_Id AND
Stock_Buffer.Cat_Id = MANEMART.dbo.Stock_Fact.Category_Id AND
Stock_Buffer.Available_Qty = MANEMART.dbo.Stock_Fact.Available_Qty
WHERE (MANEMART.dbo.Stock_Fact.Period_Id IS NULL) AND
(MANEMART.dbo.Stock_Fact.Product_Id IS NULL) AND
(MANEMART.dbo.Stock_Fact.Category_Id IS NULL)

/*INSERT INTO Internal_Consumption_Fact ERRR */
INSERT INTO MANEMART.dbo.Internal_Consumption_Fact
(Period_Id, Div_Id, Product_Id, Motive_Id, Qty, Cost)

SELECT dbo.Internal_Consumption_Buffer.Period_Id,

```

```

dbo.Internal_Consumption_Buffer.Div_Id,
dbo.Internal_Consumption_Buffer.Product_Id,
dbo.Internal_Consumption_Buffer.Motive_Id,
dbo.Internal_Consumption_Buffer.Qty,
dbo.Internal_Consumption_Buffer.Cost_Avg
FROM dbo.Internal_Consumption_Buffer LEFT OUTER JOIN
MANEMART.dbo.Internal_Consumption_Fact ON
dbo.Internal_Consumption_Buffer.Div_Id =
MANEMART.dbo.Internal_Consumption_Fact.Div_Id AND
dbo.Internal_Consumption_Buffer.Period_Id =
MANEMART.dbo.Internal_Consumption_Fact.Period_Id AND
dbo.Internal_Consumption_Buffer.Product_Id =
MANEMART.dbo.Internal_Consumption_Fact.Product_Id AND
dbo.Internal_Consumption_Buffer.Motive_Id =
MANEMART.dbo.Internal_Consumption_Fact.Motive_Id
WHERE (MANEMART.dbo.Internal_Consumption_Fact.Motive_Id IS NULL) AND
(MANEMART.dbo.Internal_Consumption_Fact.Product_Id IS NULL) AND
(MANEMART.dbo.Internal_Consumption_Fact.Div_Id IS NULL) AND
(MANEMART.dbo.Internal_Consumption_Fact.Period_Id IS NULL) AND
(dbo.Internal_Consumption_Buffer.Div_Id = 1)

/* SHIPMENT POST*/
INSERT INTO MANEMART.dbo.Shipment_Post_Fact
(Period_Id, Invoice_Id, Customer_Id, Currency_Id, Product_Id, Category_Id,
Qty, Cost, COGS)
SELECT Shipment_Post_Buffer.Period_Id, Shipment_Post_Buffer.Invoice_Id,
Shipment_Post_Buffer.Cust_Id, 1 AS Curr_Id, Shipment_Post_Buffer.Product_Id,
Shipment_Post_Buffer.Category_Id, Shipment_Post_Buffer.Qty,
Shipment_Post_Buffer.Cost, Shipment_Post_Buffer.COGS
FROM Shipment_Post_Buffer LEFT OUTER JOIN MANEMART.dbo.Shipment_Post_Fact ON
Shipment_Post_Buffer.Period_Id = MANEMART.dbo.Shipment_Post_Fact.Period_Id
AND Shipment_Post_Buffer.Invoice_Id =
MANEMART.dbo.Shipment_Post_Fact.Invoice_Id AND
Shipment_Post_Buffer.Cust_Id = MANEMART.dbo.Shipment_Post_Fact.Customer_Id
AND Shipment_Post_Buffer.Product_Id =
MANEMART.dbo.Shipment_Post_Fact.Product_Id AND
Shipment_Post_Buffer.Category_Id =
MANEMART.dbo.Shipment_Post_Fact.Category_Id
WHERE (MANEMART.dbo.Shipment_Post_Fact.Period_Id IS NULL) AND
(MANEMART.dbo.Shipment_Post_Fact.Invoice_Id IS NULL) AND
(MANEMART.dbo.Shipment_Post_Fact.Customer_Id IS NULL) AND
(MANEMART.dbo.Shipment_Post_Fact.Product_Id IS NULL) AND
(MANEMART.dbo.Shipment_Post_Fact.Category_Id IS NULL)

/* SALES ACTIVITY */
INSERT INTO MANEMART.dbo.Sales_Activity_Fact (Period_Id, Sales_Id, Cust_Id,
Area_Id, Agent_Id, Category_Id, Total_Sales)
SELECT Sales_Buffer.Period_Id, Sales_Buffer.Sales_Id, Sales_Buffer.Cust_Id,
Sales_Buffer.Area_Id, Sales_Buffer.Agent_Id, Sales_Buffer.Category_Id,
Sales_Buffer.Total_USD
FROM Sales_Buffer LEFT OUTER JOIN
MANEMART.dbo.Sales_Activity_Fact ON Sales_Buffer.Period_Id =
MANEMART.dbo.Sales_Activity_Fact.Period_Id AND
Sales_Buffer.Sales_Id = MANEMART.dbo.Sales_Activity_Fact.Sales_Id AND
Sales_Buffer.Cust_Id = MANEMART.dbo.Sales_Activity_Fact.Cust_Id
WHERE (MANEMART.dbo.Sales_Activity_Fact.Period_Id IS NULL) AND
(MANEMART.dbo.Sales_Activity_Fact.Sales_Id IS NULL) AND
(MANEMART.dbo.Sales_Activity_Fact.Cust_Id IS NULL)

```

```

/* SALES STATISTIC */
INSERT INTO MANEMART.dbo.Sales_Statistic_Fact (Period_Id, Cust_Id, State_Id,
Product_Id, Qty, Total_IDR, Total_EUR, Total_USD)
SELECT Sales_Buffer.Period_Id, Sales_Buffer.Cust_Id, 1 AS State_Id,
Sales_Buffer.Product_Id, Sales_Buffer.Qty, Sales_Buffer.Total_IDR,
Sales_Buffer.Total_EUR, Sales_Buffer.Total_USD
FROM Sales_Buffer LEFT OUTER JOIN
MANEMART.dbo.Sales_Statistic_Fact ON Sales_Buffer.Period_Id =
MANEMART.dbo.Sales_Statistic_Fact.Period_Id AND
Sales_Buffer.Cust_Id = MANEMART.dbo.Sales_Statistic_Fact.Cust_Id AND
Sales_Buffer.Product_Id = MANEMART.dbo.Sales_Statistic_Fact.Product_Id
WHERE (MANEMART.dbo.Sales_Statistic_Fact.Period_Id IS NULL) AND
(MANEMART.dbo.Sales_Statistic_Fact.Cust_Id IS NULL) AND
(MANEMART.dbo.Sales_Statistic_Fact.Product_Id IS NULL)
GO

```

#### 4.5 SISTEM PENDUKUNG

Dalam proses sinkronisasi basis data operasional menjadi *independent data mart* ini, didukung oleh perangkat lunak dan perangkat keras dengan spesifikasi sebagai berikut ini:

1. Perangkat lunak
  - Sistem operasi MS Windows XP
  - MS SQL Server 2000,
2. Perangkat keras
  - Prosesor Intel ® Core (TM) 2 Duo CPU T7300 @ 2.00 GHz
  - Memori utama 2.00 GB
  - Harddisk 120 GB
  - DVD/CD drives TEAC DW-224E-V



## BAB V

### ANALISIS IMPLEMENTASI INDEPENDENT DATA MART

Tujuan dari tahap ini adalah menganalisis hasil yang diberikan oleh implementasi *independent data mart*. Pembuktian analisis dilakukan dengan cara melakukan serangkaian pengujian berupa pengujian *query* terhadap *independent data mart* bagian *Inventory Control* dan *Sales*.

#### 5.1 PENGUJIAN

Untuk menguji akurasi data yang terdapat pada *independent data mart* bagian *Inventory Control* dan *Sales* maka dibuat serangkaian *query* untuk menampilkan permintaan data yang sering muncul di bagian *Inventory Control* dan *Sales* seperti yang disebutkan pada bab 3.5

Berikut ini merupakan pertanyaan berupa permintaan *query* dan perintah *query* yang digunakan untuk menampilkan data yang ada di *independent data mart*.

1. Transaksi stock apa saja yang dilakukan selama tahun 2007, berapa jumlah dan biaya masing-masing transaksi?

Query pada independent data mart:

```
SELECT Period.Tahun, Motive.In_Out, Motive.Motive_Desc,
SUM(Summarize_Transaction_Fact.Qty) AS Total_Qty,
SUM(Summarize_Transaction_Fact.Cost) AS Total_Cost Motive
ON Summarize_Transaction_Fact.Motive_Id = Motive.Motive_Id INNER JOIN
Period ON Summarize_Transaction_Fact.Period_Id = Period.Period_Id
GROUP BY Period.Tahun, Motive.In_Out, Motive.Motive_Desc
HAVING (Period.Tahun = 2007)
ORDER BY Motive.In_Out, Motive.Motive_Desc
```



Hasil:

	In_Out	Motive_Desc	Total_Qty	Total_Cost
1	I	Adj. FOR LOSS PROD.	36480.71995700548	1647490790.7283971
2	I	ADJUSTMENT IN	114028.74323534781	130535920.08960849
3	I	COMMUNAL	69865.2609999999999	1081234.1478274523
4	I	OTHERS IN	4270.8298634903867	1375236743.8572073
5	I	PHYSICAL CHECKING	10091.877894758014	911039840.56903696
6	I	PRODUCTION ORDER	3255191.5721000037	2276459186.4046268
7	I	RAW MATERIAL PROCESS	41807.2849999999945	1214711213.9521022
8	I	RECEPTION	3105366.2830000054	5565366100.6389885
9	I	RESTORATION	30938.9959999999999	49075934.856112018
10	I	RETURN FROM CUSTOMER	7051.8299999999999	2422622.548912873
11	O	ADJUSMENT OUT	61954.9439999999668	1312506365.8624229
12	O	COMMUNAL	69865.2609999999999	1081234.147827452
13	O	CUSTOMER DELIVERY	2922740.4580000089	3566072406.7272048
14	O	LABORATORY DELIVERY	10249.4615999999993	4408921231.8594332
15	O	LOSS (OTHER)	-2658.5999999999999	12567513.490932437
16	O	LOSS PRODUCTION	36777.483957005854	1655950502.2212453
17	O	OTHERS OUT	1370.847	23347640.584533013
18	O	PHYSICAL CHECKING	21613.187536561061	2231592125.5095301
19	O	PRODUCTION ORDER	3297526.1273143059	275278110853.65833
20	O	QC RETAIN (SAMPLE)	100.33100000000221	5448629717.8559074
21	O	RAW MATERIAL PROCESS	31505.4349999999979	817921749.16084826
22	O	REJECT	2651.5999999999995	1328371.5088169142
23	O	RESTORATION	33084.9959999999999	49912421.955954522
24	O	RETURN TO SUPPLIER	6162.0	55292.364813954831

Query pada basis data *as-is*:

```
USE MERP
SELECT PSIS_SQL.dbo.Stock_In_Out.In_Out, dbo.Motive.Motive AS
Motive_Desc, SUM(PSIS_SQL.dbo.Stock_In_Out.Qty) AS Total_Qty,
SUM(PSIS_SQL.dbo.Stock_In_Out.Cost_Avg) AS Total_Cost
FROM PSIS_SQL.dbo.Stock_In_Out INNER JOIN
dbo.Motive ON PSIS_SQL.dbo.Stock_In_Out.Motive = dbo.Motive.ID AND
PSIS_SQL.dbo.Stock_In_Out.In_Out = dbo.Motive.In_Out_Status
WHERE (YEAR(PSIS_SQL.dbo.Stock_In_Out.[Date]) = 2007)
GROUP BY PSIS_SQL.dbo.Stock_In_Out.In_Out, dbo.Motive.Motive
ORDER BY PSIS_SQL.dbo.Stock_In_Out.In_Out, dbo.Motive.Motive
```

Hasil:

	In_Out	Motive_Desc	Total_Qty	Total_Cost
1	I	Adj. FOR LOSS PROD.	36480.71995700548	1647490790.7283971
2	I	ADJUSTMENT IN	114028.74323534781	130535920.08960849
3	I	COMMUNAL	69865.2609999999999	1081234.1478274523
4	I	OTHERS IN	4270.8298634903867	1375236743.8572073
5	I	PHYSICAL CHECKING	10091.877894758014	911039840.56903696
6	I	PRODUCTION ORDER	3255191.5721000037	2276459186.4046268
7	I	RAW MATERIAL PROCESS	41807.2849999999945	1214711213.9521022
8	I	RECEPTION	3105366.2830000054	5565366100.6389885
9	I	RESTORATION	30938.9959999999999	49075934.856112018
10	I	RETURN FROM CUSTOMER	7051.8299999999999	2422622.548912873
11	O	ADJUSMENT OUT	61954.9439999999668	1312506365.8624229
12	O	COMMUNAL	69865.2609999999999	1081234.147827452
13	O	CUSTOMER DELIVERY	2922740.4580000089	3566072406.7272048
14	O	LABORATORY DELIVERY	10249.4615999999993	4408921231.8594332
15	O	LOSS (OTHER)	-2658.5999999999999	12567513.490932437
16	O	LOSS PRODUCTION	36777.483957005854	1655950502.2212453
17	O	OTHERS OUT	1370.847	23347640.584533013
18	O	PHYSICAL CHECKING	21613.187536561061	2231592125.5095301
19	O	PRODUCTION ORDER	3297526.1273143059	275278110853.65833
20	O	QC RETAIN (SAMPLE)	100.33100000000221	5448629717.8559074
21	O	RAW MATERIAL PROCESS	31505.4349999999979	817921749.16084826
22	O	REJECT	2651.5999999999995	1328371.5088169142
23	O	RESTORATION	33084.9959999999999	49912421.955954522
24	O	RETURN TO SUPPLIER	6162.0	55292.364813954831

2. Berapa jumlah dan biaya transaksi stock untuk Lab divisi Flavour selama tahun 2007?

Query pada *independent data mart*:

```
SELECT Period.Tahun, Motive.Motive_Desc, Division.Division_Name,
SUM(Internal_Consumption_Fact.Qty) AS Total_Qty,
SUM(Internal_Consumption_Fact.Cost) AS Total_Cost
FROM Internal_Consumption_Fact INNER JOIN Motive
ON Internal_Consumption_Fact.Motive_Id = Motive.Motive_Id
INNER JOIN Product
ON Internal_Consumption_Fact.Period_Id = Product.Product_Id
INNER JOIN Period
ON Internal_Consumption_Fact.Period_Id = Period.Period_Id
INNER JOIN Division ON Internal_Consumption_Fact.Div_Id =
Division.Division_Id
WHERE (Period.Tahun = 2007) AND (Motive.N_Motive = 'i')
GROUP BY Period.Tahun, Motive.Motive_Desc, Division.Division_Name
HAVING (Division.Division_Name = N'flavour')
```

Hasil:

	Tahun	Motive_Desc	Division_Name	Total_Qty	Total_Cost
1	2007	LABORATORY DELIVERY	Flavour	8036.99390000000059	2951664023.5535779

Query pada basis data as-is:

```
USE MERP

SELECT YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]) AS Tahun, dbo.Motive.Motive
AS Motive_Description,
SysSQLClone.dbo.Product_Identification.Responsible AS Division_Name,
SUM(Psis_SQL.dbo.Stock_In_Out.Qty) AS Total_Qty,
SUM(Psis_SQL.dbo.Stock_In_Out.Cost_Avg) AS Total_Cost
FROM Psis_SQL.dbo.Stock_In_Out INNER JOIN
dbo.Motive ON Psis_SQL.dbo.Stock_In_Out.Motive = dbo.Motive.ID AND
Psis_SQL.dbo.Stock_In_Out.In_Out = dbo.Motive.In_Out_Status INNER JOIN
SysSQLClone.dbo.Product_Identification ON
Psis_SQL.dbo.Stock_In_Out.Internal_Ref =
SysSQLClone.dbo.Product_Identification.Reference
```

```

GROUP BY dbo.Motive.Motive, YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]),
SysSQLClone.dbo.Product_Identification.Responsible
HAVING (YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]) = 2007) AND
(dbo.Motive.Motive = 'i') AND
(SysSQLClone.dbo.Product_Identification.Responsible = N'FLV')
ORDER BY SysSQLClone.dbo.Product_Identification.Responsible

```

Hasil:

	Tahun	Motive_Desc	Division_Name	Total_Qty	Total_Cost
1	2007	LABORATORY DELIVERY	Flavour	8036.99390000000059	2951664023.5535779

3. Berapa jumlah dan biaya transaksi stock untuk *QC retain sample* selama tahun 2007?

Query pada *independent data mart*:

```

SELECT Period.Tahun AS YEAR, Motive.Motive_Desc, Division.Division_Name,
SUM(Internal_Consumption_Fact.Qty) AS Total_Qty,
SUM(Internal_Consumption_Fact.Cost) AS Total_Cost
FROM Internal_Consumption_Fact INNER JOIN
Motive ON Internal_Consumption_Fact.Motive_Id = Motive.Motive_Id INNER
JOIN Product
ON Internal_Consumption_Fact.Product_Id = Product.Product_Id INNER JOIN
Period ON Internal_Consumption_Fact.Period_Id = Period.Period_Id INNER
JOIN Division ON Internal_Consumption_Fact.Div_Id = Division.Division_Id
WHERE (Period.Tahun = 2007) AND (Motive.N_Motive = 'm')
GROUP BY Period.Tahun, Motive.Motive_Desc, Division.Division_Name

```

Hasil:

	YEAR	Motive_Desc	Division_Name	Total_Qty	Total_Cost
1	2007	QC RETAIN (SAMPLE)	Flavour	62.743999999999943	4338864530.0277996
2	2007	QC RETAIN (SAMPLE)	Fragrance	37.586999999999954	1109765187.8281076

Query pada basis data *as-is*:

```
USE MERP
```

```

SELECT YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]) AS Tahun, dbo.Motive.Motive
AS Motive_Description,
SysSQLClone.dbo.Product_Identification.Responsible AS Division_Name,
SUM(Psis_SQL.dbo.Stock_In_Out.Qty) AS Total_Qty,
SUM(Psis_SQL.dbo.Stock_In_Out.Cost_Avg) AS Total_Cost
FROM Psis_SQL.dbo.Stock_In_Out INNER JOIN
dbo.Motive ON Psis_SQL.dbo.Stock_In_Out.Motive = dbo.Motive.ID AND
Psis_SQL.dbo.Stock_In_Out.In_Out = dbo.Motive.In_Out_Status INNER JOIN
SysSQLClone.dbo.Product_Identification ON
Psis_SQL.dbo.Stock_In_Out.Internal_Ref =
SysSQLClone.dbo.Product_Identification.Reference
GROUP BY dbo.Motive.Motive, YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]),
SysSQLClone.dbo.Product_Identification.Responsible
HAVING (YEAR(Psis_SQL.dbo.Stock_In_Out.[Date]) = 2007) AND
(dbo.Motive.Motive = 'M')
ORDER BY SysSQLClone.dbo.Product_Identification.Responsible

```

Hasil:

	YEAR	Motive_Desc	Division_Name	Total_Qty	Total_Cost
1	2007	QC RETAIN (SAMPLE)	Flavour	62.743999999999943	4338864530.0277996
2	2007	QC RETAIN (SAMPLE)	Fragrance	37.586999999999954	1109765187.8281076

4. Berapa jumlah stock per kategori barang, dan harga jual barang kepada pelanggan pada bulan januari 2007?

Query:

```

SELECT Period.Tahun, Period.Bulan, Category_Jual.Category_Desc,
SUM(Shipment_Post_Fact.Qty) AS Total_Qty, SUM(Shipment_Post_Fact.Cost)
AS Total_Cost, SUM(Shipment_Post_Fact.COGS) AS Total_COGS
FROM Shipment_Post_Fact INNER JOIN Period ON
Shipment_Post_Fact.Period_Id = Period.Period_Id INNER JOIN Currency ON
Shipment_Post_Fact.Currency_Id = Currency.Currency_Id INNER JOIN
Category_Jual ON Shipment_Post_Fact.Category_Id =

```

```

Category_Jual.Category_Id
GROUP BY Period.Tahun, Period.Bulan, Category_Jual.Category_Desc
HAVING (Period.Tahun = 2007) AND (Period.Bulan = 1)

```

### Hasil:

	Tahun	Bulan	Category_Desc	Total_Qty	Total Cost	Total COGS
1	2007	1	Food Flavour Savoury	85620.0	164060232.10545447	56814990.914869994
2	2007	1	Food Flavour Sweet	5804056.4000000171	2164352706.7221117	820635132.21391666
3	2007	1	Oral Care Flav.	53792.0	61550456.387088589	22390201.282287244
4	2007	1	Perf. Comp.	1725890.7	3427598971.680829	1266299926.3845332
5	2007	1	RM Food Flav.	271323.48000000074	2821363742.5167432	2438251545.7172928
6	2007	1	RM Frag	34340.000000000051	252797227.95538217	200426581.06374651
7	2007	1	RM Oral Care	3375.0	35656945.199999973	25927055.99100006
8	2007	1	RM Tob. Flav.	8793.5	422297530.63399982	219214664.72058246
9	2007	1	Tob. Flav.	21643.0	54881256.386666708	23179640.476066913

### Query pada basis data *as-is*:

```

USE MERP
SELECT (PSIS_SQL.dbo.Stock_In_Out.[Date]) AS Tahun,
MONTH(PSIS_SQL.dbo.Stock_In_Out.[Date]) AS Bulan,
PBookSQL.dbo.Category.Description AS Category,
SUM(IC.dbo.Invoice_Detail_IC.Qty) AS Total_Qty,
SUM(IC.dbo.Invoice_Detail_IC.SellingPrice)
AS Total_Cost, SUM(PSIS_SQL.dbo.Stock_In_Out.Cost_Avg) AS Total_COGS
FROM IC.dbo.Invoice_Detail_IC INNER JOIN
EXIS.dbo.DO ON IC.dbo.Invoice_Detail_IC.Dos_Ref = EXIS.dbo.DO.DO_Ref
INNER JOIN
PBookSQL.dbo.External_List ON IC.dbo.Invoice_Detail_IC.Internal_Ref =
PBookSQL.dbo.External_List.Internal_Ref AND
IC.dbo.Invoice_Detail_IC.External_Ref =
PBookSQL.dbo.External_List.External_Ref AND
IC.dbo.Invoice_Detail_IC.Category = PBookSQL.dbo.External_List.Category
INNER JOIN
PSIS_SQL.dbo.Stock_In_Out ON PBookSQL.dbo.External_List.Internal_Ref =
PSIS_SQL.dbo.Stock_In_Out.Internal_Ref AND
IC.dbo.Invoice_Detail_IC.Lot_No = PSIS_SQL.dbo.Stock_In_Out.Lot_No
INNER JOIN
PBookSQL.dbo.Category ON PBookSQL.dbo.External_List.Category =
PBookSQL.dbo.Category.Category
GROUP BY MONTH(PSIS_SQL.dbo.Stock_In_Out.[Date]),
YEAR(PSIS_SQL.dbo.Stock_In_Out.[Date]),
PBookSQL.dbo.Category.Description
HAVING (YEAR(PSIS_SQL.dbo.Stock_In_Out.[Date]) = 2007) AND
(MONTH(PSIS_SQL.dbo.Stock_In_Out.[Date]) = 1)
ORDER BY PBookSQL.dbo.Category.Description

```

Hasil:

	Tahun	Bulan	Category_Desc	Total_Qty	Total_Cost	Total_COGS
1	2007	1	Food Flavour Savoury	85620.0	164060232.10545447	56814990.914869994
2	2007	1	Food Flavour Sweet	5804056.4000000171	2164352706.7221117	820635132.21391666
3	2007	1	Oral Care Flav.	53792.0	61550456.387088589	22390201.282287244
4	2007	1	Perf. Comp.	1725890.7	3427598971.680829	1266299926.3845332
5	2007	1	RM Food Flav.	271323.48000000074	2821363742.5167432	2438251545.7172928
6	2007	1	RM Frag	34340.000000000051	252797227.95538217	200426581.06374651
7	2007	1	RM Oral Care	3375.0	35656945.199999973	25927055.99100006
8	2007	1	RM Tob. Flav.	8793.5	422297530.63399982	219214664.72058246
9	2007	1	Tob. Flav.	21643.0	54881256.386666708	23179640.476066913

5. Berapa angka penjualan masing-masing *sales person* pada masing-masing area di bulan januari 2008?

Query:

```
SELECT Sales.Full_Name, Area.Area_Name,
SUM(Sales_Activity_Fact.Total_Sales) AS Total_Sales
FROM Sales_Activity_Fact INNER JOIN
Area ON Sales_Activity_Fact.Area_Id = Area.Area_Id INNER JOIN
Period ON Sales_Activity_Fact.Period_Id = Period.Period_Id INNER JOIN
Sales ON Sales_Activity_Fact.Sales_Id = Sales.Sales_Id
WHERE (Period.Tahun = 2008) AND (Period.Bulan = 1)
GROUP BY Sales.Full_Name, Area.Area_Name
ORDER BY Sales.Full_Name, Area.Area_Name
```

Hasil:

	Full_Name	Area_Name	Total_Sales
1	Ay LING	CENTRAL JAVA	92527.5
2	Ay LING	WEST JAVA	286377.78130000003
3	Bernard J LEYNAUD	BALI	600.09090000000003
4	Bernard J LEYNAUD	EAST JAVA	123849.53870000002
5	Bernard J LEYNAUD	SUMATRA	64986.075700000001
6	Bernard J LEYNAUD	WEST JAVA	118663.48
7	Didier PARIZOT	CENTRAL JAVA	3061.9101999999998
8	Didier PARIZOT	EAST JAVA	29103.625600000003
9	Didier PARIZOT	SUMATRA	63036.0
10	Didier PARIZOT	WEST JAVA	39510.601300000002
11	Disman STEVANUS	CENTRAL JAVA	475.577
12	Disman STEVANUS	EAST JAVA	24640.154500000001
13	Disman STEVANUS	SUMATRA	25851.5903999999994
14	Disman STEVANUS	WEST JAVA	177230.4914
15	Evi SANTIOSO	EAST JAVA	2000.0
16	Evi SANTIOSO	SULAWESI	2299.5455000000002

Query pada basis data *as-is*:

```

SELECT SysSQLClone.dbo.SecUser.FirstName +
SysSQLClone.dbo.SecUser.LastName AS Full_Name,
SysSQLClone.dbo.Client_Area.Area_Name,
SUM(AR_SQL.dbo.Montly_Rekap.TotalUSD) AS Total_Sales
FROM AR_SQL.dbo.Montly_Rekap INNER JOIN
SysSQLClone.dbo.Client ON AR_SQL.dbo.Montly_Rekap.Customer_Ref =
SysSQLClone.dbo.Client.N_Client INNER JOIN
PBookSQL.dbo.External_List ON AR_SQL.dbo.Montly_Rekap.Customer_Ref =
PBookSQL.dbo.External_List.Customer_Ref AND
AR_SQL.dbo.Montly_Rekap.Internal_Ref =
PBookSQL.dbo.External_List.Internal_Ref AND
AR_SQL.dbo.Montly_Rekap.Internal_Ref =
PBookSQL.dbo.External_List.External_Ref INNER JOIN
SysSQLClone.dbo.Client_Address ON SysSQLClone.dbo.Client.Region =
SysSQLClone.dbo.Client_Address.Region INNER JOIN
SysSQLClone.dbo.Client_Area ON SysSQLClone.dbo.Client_Address.Area =
SysSQLClone.dbo.Client_Area.Code_Area INNER JOIN
SysSQLClone.dbo.SecUser ON PBookSQL.dbo.External_List.Sales_Person =
SysSQLClone.dbo.SecUser.UserID
WHERE (AR_SQL.dbo.Montly_Rekap.[Year] = '2008') AND
(AR_SQL.dbo.Montly_Rekap.Period = 1)
GROUP BY SysSQLClone.dbo.SecUser.FirstName +
SysSQLClone.dbo.SecUser.LastName, SysSQLClone.dbo.Client_Area.Area_Name

```

Hasil:

	Full_Name	Area_Name	Total_Sales
1	Ay LING	CENTRAL JAVA	92527.5
2	Ay LING	WEST JAVA	286377.781300000003
3	Bernard J LEYNAUD	BALI	600.090900000000003
4	Bernard J LEYNAUD	EAST JAVA	123849.538700000002
5	Bernard J LEYNAUD	SUMATRA	64986.0757000000001
6	Bernard J LEYNAUD	WEST JAVA	118663.48
7	Didier PARIZOT	CENTRAL JAVA	3061.91019999999998
8	Didier PARIZOT	EAST JAVA	29103.6256000000003
9	Didier PARIZOT	SUMATRA	63036.0
10	Didier PARIZOT	WEST JAVA	39510.6013000000002
11	Disman STEVANUS	CENTRAL JAVA	475.577
12	Disman STEVANUS	EAST JAVA	24640.1545000000001
13	Disman STEVANUS	SUMATRA	25851.5903999999994
14	Disman STEVANUS	WEST JAVA	177230.4914
15	Evi SANTIOSO	EAST JAVA	2000.0
16	Evi SANTIOSO	SULAWESI	2299.54550000000002

6. Berapa angka statistik penjualan per produk bulan januari 2008?

Query pada *independent data mart*:

```

SELECT Product.Internal_Ref, Product.Product_Name,
SUM(Sales_Statistic_Fact.Qty) AS Total_Qty,
SUM(Sales_Statistic_Fact.Total_IDR) AS Total_IDR,
SUM(Sales_Statistic_Fact.Total_EUR) AS Total_EUR,
SUM(Sales_Statistic_Fact.Total_USD) AS Total_USD
FROM Sales_Statistic_Fact INNER JOIN

```

```

Period ON Sales_Statistic_Fact.Period_Id = Period.Period_Id INNER JOIN
Product ON Sales_Statistic_Fact.Product_Id = Product.Product_Id
WHERE (Period.Tahun = 2008) AND (Period.Bulan = 1)
GROUP BY Product.Internal_Ref, Product.Product_Name
ORDER BY Product.Internal_Ref

```

### Hasil:

	Internal_Ref	Product_Name	Total_Qty	Total_IDR	Total_EUR	Total_USD
1	23271	YLANG OLF 741 P99 (Profor BsL)	15.0	7242226.0	522.0	767.18270000000007
2	302300	ROSAFIX(R58746)	25.0	8934026.0	642.211300000000005	944.5
3	302449	ANISEED EXTRA	90.0	33360998.0	2397.80110000000001	3527.5
4	304910	CLOVE ZANZIBAR No2	500.0	221186700.0	16202.6463	23500.0
5	307293	PEPPERMINT (MI)	25.0	18198950.0	1307.6739	1925.0
6	309645	CINNAMON OIL	25.0	13600629.0	996.290399999999998	1445.0
7	309695	ZANZIBAR	50.0	23329800.0	1683.49929999999999	2474.5119
8	314681	PEPPERMINT (MI MOD)	50.0	17883180.0	1310.00119999999999	1900.0
9	36982	PAZEA 99117	25.0	6019364.0	432.694500000000001	636.363600000000002
10	37388	SIENNE GREEN 94018 (MI MOD)	50.0	10528318.0	756.505599999999996	1113.636400000000001
11	40147	YUCCA LINOTE 99242 (MI)	100.0	12788896.0	926.569100000000005	1360.0
12	40527	BERENICE 99612	25.0	13647690.0	999.737799999999999	1450.0
13	40534	BERENICE 99619	25.0	4490650.0	322.6728	475.0
14	40767	OKAPI 99835	25.0	2687216.0	193.167200000000001	284.090899999999998
15	41074	APPLE 100091	50.0	7522880.0	545.040700000000002	800.0
16	45028	SPITZBERG BLACK BENTHLEY GC...	600.0	87564073.0	6355.36650000000001	9300.0

### Query pada basis *data as-is*:

```

SELECT AR_SQL.dbo.Montly_Rekap.Internal_Ref,
SysSQLClone.dbo.Product_Identification.ProductName,
SUM(AR_SQL.dbo.Montly_Rekap.Quantity_IO) AS Total_Qty,
SUM(AR_SQL.dbo.Montly_Rekap.Total) AS Total_IDR,
SUM(AR_SQL.dbo.Montly_Rekap.TotalFRF) AS Total_EUR,
SUM(AR_SQL.dbo.Montly_Rekap.TotalUSD) AS Total_USD
FROM AR_SQL.dbo.Montly_Rekap INNER JOIN
SysSQLClone.dbo.Product_Identification
ON AR_SQL.dbo.Montly_Rekap.Internal_Ref =
SysSQLClone.dbo.Product_Identification.Reference
WHERE (AR_SQL.dbo.Montly_Rekap.[Year] = '2008') AND
(AR_SQL.dbo.Montly_Rekap.Period = 1)
GROUP BY AR_SQL.dbo.Montly_Rekap.Internal_Ref,
SysSQLClone.dbo.Product_Identification.ProductName
ORDER BY AR_SQL.dbo.Montly_Rekap.Internal_Ref

```



Hasil:

	Internal_Ref	Product_Name	Total_Qty	Total_IDR	Total_EUR	Total_USD
1	23271	YLANG OLF 741 P99 (Profor BsL)	15.0	7242226.0	522.0	767.18270000000007
2	302300	ROSAFIX(R58746)	25.0	8934026.0	642.211300000000005	944.5
3	302449	ANISEED EXTRA	90.0	33360998.0	2397.80110000000001	3527.5
4	304910	CLOVE ZANZIBAR No2	500.0	221186700.0	16202.6463	23500.0
5	307293	PEPPERHINT (MI)	25.0	18198950.0	1307.6739	1925.0
6	309645	CINNAMON OIL	25.0	13600629.0	996.29039999999998	1445.0
7	309695	ZANZIBAR	50.0	23329800.0	1683.4992999999999	2474.5119
8	314681	PEPPERHINT (MI MOD)	50.0	17883180.0	1310.0011999999999	1900.0
9	36982	PAZE 99117	25.0	6019364.0	432.69450000000001	636.363600000000002
10	37388	SIENNE GREEN 94018 (MI MOD)	50.0	10528318.0	756.50559999999996	1113.63640000000001
11	40147	YUCCA LINOTE 99242 (MI)	100.0	12788896.0	926.569100000000005	1360.0
12	40527	BERENICE 99612	25.0	13647690.0	999.73779999999999	1450.0
13	40534	BERENICE 99619	25.0	4490650.0	322.6728	475.0
14	40767	OKAPI 99835	25.0	2687216.0	193.167200000000001	284.09089999999998
15	41074	APPLE 100091	50.0	7522880.0	545.040700000000002	800.0
16	45028	SPITZBERG BLACK BENTHLEY GC...	600.0	87564073.0	6355.36650000000001	9300.0

7. Berapa statistik penjualan per *customer* di bulan januari tahun 2008?

Query pada *independent data mart*:

```
SELECT Customer.Customer_Name, SUM(Sales_Statistic_Fact.Qty) AS
Total_Qty, SUM(Sales_Statistic_Fact.Total_IDR) AS Total_IDR,
SUM(Sales_Statistic_Fact.Total_EUR) AS Total_EUR,
SUM(Sales_Statistic_Fact.Total_USD) AS Total_USD
FROM Sales_Statistic_Fact INNER JOIN
Period ON Sales_Statistic_Fact.Period_Id = Period.Period_Id INNER JOIN
Customer ON Sales_Statistic_Fact.Cust_Id = Customer.Cust_Id
WHERE (Period.Tahun = 2008) AND (Period.Bulan = 1)
GROUP BY Customer.Customer_Name
ORDER BY Customer.Customer_Name
```

Hasil:

	Customer_Name	Total_Qty	Total_IDR	Total_EUR	Total_USD
1	ABC PRESIDENT INDONESIA, PT	2500.0	269581500.0	19378.52900000000002	28500.0
2	ACUNG, MR	25.0	5374432.0	386.33429999999998	568.18179999999995
3	ADIMULIA SARIMAS INDUSTRI, PT	300.0	16458660.0	1183.11020000000001	1740.0
4	AGUS, MR/LANGGENG PANGAN MAKHUR , PT	575.0	92352236.0	6765.10220000000003	9811.9712
5	AL - HARUM	75.0	42585386.0	3097.05490000000001	4532.39709999999992
6	AL-AMIN, UD/ LA-PANTA	25.0	3213000.0	230.8681	339.856100000000003
7	ALAM MEGAH JAYA, UD	240.0	24455200.0	1764.1495	2592.86210000000003
8	AMCHEM MARKETING	4495.0	1014263043.0	73384.1806000000007	107748.950000000001
9	ANEKA BOGA CITRA, PT	100.0	8697680.0	624.96619999999996	920.0
10	ANEKA COFFEE INDUSTRY, PT	5.0	2079880.0	149.44839999999999	220.0
11	ANEKA COSMETIC, PT	25.0	5770391.0	418.071000000000003	613.63639999999998
12	ANEKA KARUNIA/ WILSON, MR	1375.0	102282007.0	7415.7369999999995	10837.5
13	ANEKA KREASI PANGAN, PT/HARDJA TARU...	50.0	5683612.0	413.09339999999997	605.0
14	ANUGERAH SEJAHTERA NASIONAL, PT	100.0	15040000.0	1101.729	1597.9260999999999
15	ANUGRAH JAYA SENTOSA, PT	50.0	4297273.0	308.777800000000001	454.5455
16	ARCA SALES & MARKETING	100.0	29729637.0	2137.0778	3143.0

Query pada basis data *as-is*:

```

SELECT SysSQLClone.dbo.Client.ClientName AS Customer_Name,
SUM(AR_SQL.dbo.Montly_Rekap.Quantity_IO) AS Total_Qty,
SUM(AR_SQL.dbo.Montly_Rekap.Total) AS Total_IDR,
SUM(AR_SQL.dbo.Montly_Rekap.TotalFRF) AS Total_EUR,
SUM(AR_SQL.dbo.Montly_Rekap.TotalUSD) AS Total_USD
FROM AR_SQL.dbo.Montly_Rekap INNER JOIN
SysSQLClone.dbo.Client ON AR_SQL.dbo.Montly_Rekap.Customer_Ref =
SysSQLClone.dbo.Client.N_Client
WHERE (AR_SQL.dbo.Montly_Rekap.[Year] = '2008') AND
(AR_SQL.dbo.Montly_Rekap.Period = 1)
GROUP BY SysSQLClone.dbo.Client.ClientName
ORDER BY SysSQLClone.dbo.Client.ClientName

```

## Hasil:

	Customer Name	Total Qty	Total IDR	Total EUR	Total USD
1	ABC PRESIDENT INDONESIA, PT	2500.0	269581500.0	19378.5290000000002	28500.0
2	ACUNG, MR	25.0	5374432.0	386.334299999999998	568.181799999999995
3	ADIMULIA SARIMAS INDUSTRI, PT	300.0	16458660.0	1183.110200000000001	1740.0
4	AGUS, MR/LANGGENG PANGAN MAKMUR , PT	575.0	92352236.0	6765.102200000000003	9811.9712
5	AL - HARUM	75.0	42585386.0	3097.054900000000001	4532.397099999999992
6	AL-AMIN, UD/ LA-PANTA	25.0	3213000.0	230.8681	339.856100000000003
7	ALAM MEGAH JAYA, UD	240.0	24455200.0	1764.1495	2592.862100000000003
8	AMCHEM MARKETING	4495.0	1014263043.0	73384.180600000000007	107748.9500000000001
9	ANEKA BOGA CITRA, PT	100.0	8697680.0	624.966199999999996	920.0
10	ANEKA COFFEE INDUSTRY, PT	5.0	2079880.0	149.448399999999999	220.0
11	ANEKA COSMETIC, PT	25.0	5770391.0	418.071000000000003	613.636399999999998
12	ANEKA KARUNIA/ WILSON, MR	1375.0	102282007.0	7415.736699999999995	10837.5
13	ANEKA KREASI PANGAN, PT/HARDJA TARU...	50.0	5683612.0	413.093399999999997	605.0
14	ANUGERAH SEJAHTERA NASIONAL, PT	100.0	15040000.0	1101.729	1597.926099999999999
15	ANUGRAH JAYA SENTOSA, PT	50.0	4297273.0	308.777800000000001	454.5455
16	ARCA SALES & MARKETING	100.0	29729637.0	2137.0778	3143.0

## 5.2 ANALISIS PENGUJIAN

Dari serangkaian pengujian yang dilakukan terhadap *independent data mart* bagian *Inventory Control* dan *Sales*, dapat dilihat bahwa:

1. Relasi antar tabel fakta dan dimensi terbentuk sudah sesuai dengan tujuan, dibuktikan dengan *query* berdasarkan beberapa kebutuhan data yang sering ditanyakan ke bagian *Inventory Control* dan *Sales* yang diuji pada bab 5.2
2. Dari hasil yang ditampilkan oleh *query* yang dibuat pada tahap pengujian dapat dilihat bahwa data hasil *query* sesuai dengan yang diharapkan
3. Dari segi proses pengambilan data, pengambilan data dari *independent data mart* lebih efisien, karena akses yang dilakukan hanya terhadap *single database* sedangkan pengambilan data pada sistem *as-is* mengacu ke beberapa basis data
4. Perintah *query* yang digunakan untuk mendapatkan informasi yang diminta sesuai dengan kasus yang diujikan sesuai dengan yang diharapkan.

## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Kesimpulan tugas akhir ini adalah sebagai berikut:

1. Pembangunan *independent data mart* bagian *Inventory Control* dan *Sales* di PT Mane Indonesia dibangun untuk mengatasi masalah penyajian informasi yang berasal dari beberapa basis data operasional yang belum terintegrasi
2. Pemindahan data dari beberapa basis data operasional ke *independent data mart* terwujud dengan proses ETL (*Extract, Transformation, Loading*)
3. Informasi yang dihasilkan oleh *independent data mart* dapat menjawab kebutuhan informasi bagian *Inventory Control* dan *Sales* PT Mane Indonesia

#### 6.2 Saran

Untuk keperluan pengembangan *independent data mart* ada beberapa hal yang disarankan:

1. *Independent data mart* yang dibangun suatu saat dapat dikembangkan untuk divisi lain yang ada di PT Mane Indonesia. Perlu dilakukan analisis lebih lanjut untuk mengembangkan perancangan yang sudah ada agar mendukung pembangunan *independent data mart* pada keseluruhan divisi yang membutuhkan.

2. Melakukan evaluasi terhadap independent data mart secara berkala untuk menjamin kehandalan independent data mart sehingga sesuai dengan perubahan kebutuhan pengguna pada perusahaan

## DAFTAR REFERENSI

Inmon, W.H. Building The Data Warehouse, Third Edition. John Wiley & Sons, Inc. 2002

Connolly, Thomas; Begg Carolyn Database Systems, Fourth Edition. Addison Wesley. 2005

Humphries, Mark; Hawkins, Michael W.; Dy, Michelle C. Data Warehousing, Architecture and Implementation. Prentice Hall, Upper Saddle River, New Jersey. 1999

Purnomo, Setiawan Hari; Zulkieflimansyah Manajemen Strategi: Sebuah Konsep Pengantar. Fakultas Ekonomi Universitas Indonesia, Jakarta. 1999

<http://www.datawarehouse.com/iknowledge/articles/article.cfm>. Fuller, David R. The Fundamental of Data Warehousing: Beware of Stovepipes. 2002

<http://www.tdan.com/i014hy01.htm>. Marco, David. Migrating From Independent Data Marts. November 2000

<http://www.mimno.com/extracting-data.html>. Mimno, Myers & Holum. 2001

<http://www.dmreview.com>. Inmon, Bill. An Open Letter to Quentin Hardy. 2004

<http://www.wisetome.com/technology/database/history-database-systems>. Vyoma. History of Database Systems. 2007

[http://download.oracle.com/docs/cd/E10352\\_01/doc/bi.1013/e10312/dm\\_concepts.htm](http://download.oracle.com/docs/cd/E10352_01/doc/bi.1013/e10312/dm_concepts.htm). A Data Mart Concepts. 2008