# BAB 2
# TINJAUAN PUSTAKA

## 2.1 Profil PT Perkebunan Nusantara (PTPN) VIII Kebun Ciater

Indonesia merupakan salah satu Negara eks jajahan Belanda. Bangsa Belanda datang ke Indonesia sebelum perang dunia ke I, selama menjajah Indonesia, Belanda mendirikan perusahaan perkebunan kopi dan perkebunan kina, pada tahun 1915 Bangsa Belanda mulai membudidayakan tanaman teh yang sangat luas arealnya sehingga produksi teh sangat berlimpah maka pada tahun 1920 s/d 1922 Bangsa Belanda mulailah mendirikan pabrik untuk pengolahan teh di tiap-tiap perkebunan yang ada di Jawa Barat tak terkecuali perkebunan yang ada di kabupaten subang yaitu pabrik pengolahan teh tambakan, kasomalang, dan serengsari.Pada tahun 1934 pabrik teh didaerah Ciater didirikan dan dioperasikan pada tahun 1937 dengan kapasitas olah ± 900 ton teh kering setahun. Pada tahun 1945 Indonesia merdeka terbebas dari segala bentuk penjajahan dan pada sekitar tahun 1950-an Belanda meninggalkan Indonesia, semua peninggalan Belanda diambil alih oleh Inggris salah satunya perusahaan P&T Land PT (Pamanukan dan Tjiasem Land) yang berkantor pusat di Kota Subang. Setelah Inggris meninggalkan Indonesia semua asset diambil oleh Bangsa Indonesia. Awal mula perkebunan dikelola oleh suatu perusahaan yang diberi nama PTP. XIII, dan sekitar tahun 1979 semua peralatan termasuk mesin-mesin pengolahan diperbaiki dan dimodifikasi disesuaikan dengan kebutuhan dan perkembangan jaman.

Perkebunan ciater merupakan salah satu perkebunan yang banyak menghasilkan produksi. Karena banyaknya produksi, sehingga proses pengolahan tidak berjalan dengan baik akibat kapasitas pengolahan waktu itu hanya sedikit, maka pada tahun 1989 direksi PTP. XIII memutuskan untuk membuat pabrik dengan kapasitas pengolahan yang lebih banyak (± 60-70 ton/hari basah). Pabrik yang baru dibangun tahun 1990 diatas tanah seluas 20.000 $m^3$ dengan ketinggian dari permukaan laut ± 1050 Mtr dan suhu rata-rata 18-25 °C mampu bersaing dengan perkebunan-perkebunan lainnya dan mampu menghasilkan teh jadi yang berkualitas baik sehingga terkenal ke mancanegara.

### 2.1.1 Logo PT Perkebunan Nusantara (PTPN) VIII Kebun Ciater

Logo dari PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater dapat dilihat pada gambar dibawah ini.



**Gambar 2.1 Logo PTPN VIII Kebun Ciater**

### 2.1.2 Visi dan Misi PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater

Visi dan misi dari PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater adalah sebagai berikut:

1. Visi

   Visi dari PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater saat ini adalah:

   "Menjadi Perusahaan Agribisnis terkemuka dan terpercaya, mengutamakan kepuasan pelanggan dan kepedulian lingkungan dengan didukung oleh SDM yang profesional".

2. Misi

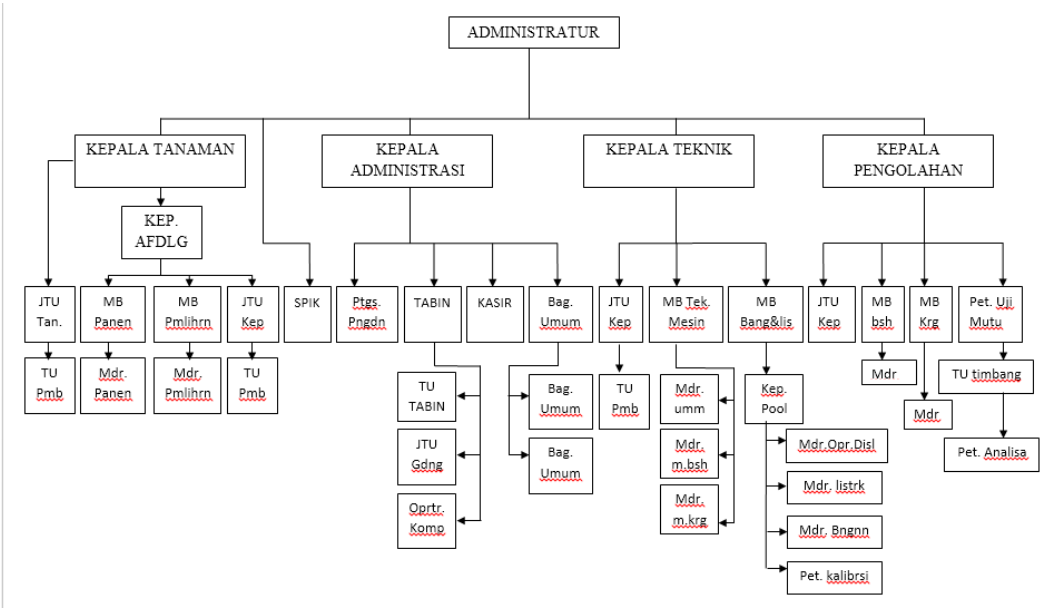
   Misi dari PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater adalah sebagai berikut:

   1. Menghasilkan produk bermutu dan ramah lingkungan yang dibutuhkan oleh pasar dan mempunyai nilai tambah tinggi.
   2. Mengelola perusahaan dengan menerapkan Good Governance dan Strong Leadership, memosisikan sumber daya manusia sebagai mitra utama, serta mengedepankan kesejahteraan karyawan melalui kesehatan perusahaan.

3. Mengoptimalkan seluruh sumber daya untuk dapat meraih peluang-peluang pengembangan bisnis, secara mandiri maupun bersama-sama mitra strategis.
4. Mengedepankan Corporate Sosial Responsibility (CSR) seiring dengan kemajuan perusahaan.

### 2.1.3 Struktur Organisasi PTPN VIII Kebun Ciater

Susunan organisasi PT. Perkebunan Nusantara (PTPN) VIII Kebun Ciater dapat dilihat pada gambar**Error! Reference source not found.** dibawah ini.




**Gambar 2.2 Struktur Organisasi PTPN VIII Kebun Ciater**

PT. Perkebunan Nusantara (PTPN) VIII Perkebunan Ciater dikepalai oleh seorang Administratur yang membawahi 4 orang kepala, yaitu Kepala Tanaman (membawahi 6 orang kepala afdeling), Kepala Administrasi, Kepala Teknik dan Kepala Pengolahan.

## 2.2 Landasan Teori

### 2.2.1 Data Warehouse

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

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

1. *Subject oriented* (berorientasi subjek)

   Data *warehouse* dirancang untuk membantu *user* dalam pengambilan keputusan. Contohnya untuk mengetahui tentang data penjualan perusahaan. Kita bisa membangun data *warehouse* yang berfokus pada penjualan. Dengan menggunakan data *warehouse*, kita bisa menjawab pertanyaan seperti "Siapa pembeli terbaik untuk barang ini tahun lalu?". Kemampuan untuk mendefinisikan sebuah data *warehouse* sebagai sebuah subjek, dalam hal ini penjualan, membuat data *warehouse subject oriented*.

2. *Integrated* (terintegrasi)

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

3. *Time-variant*

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

4. *Non-volatile*

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

Data *warehouse* dibuat untuk melayani *user* (*analyst* dan pengambil keputusan). Sehingga data *warehouse* wajib dirancang sesuai dengan persyaratan [1] berikut :

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

Sebagian besar data *warehouse* yang sukses selain memenuhi persyaratan di atas juga memiliki beberapa karakteristik [1] seperti :

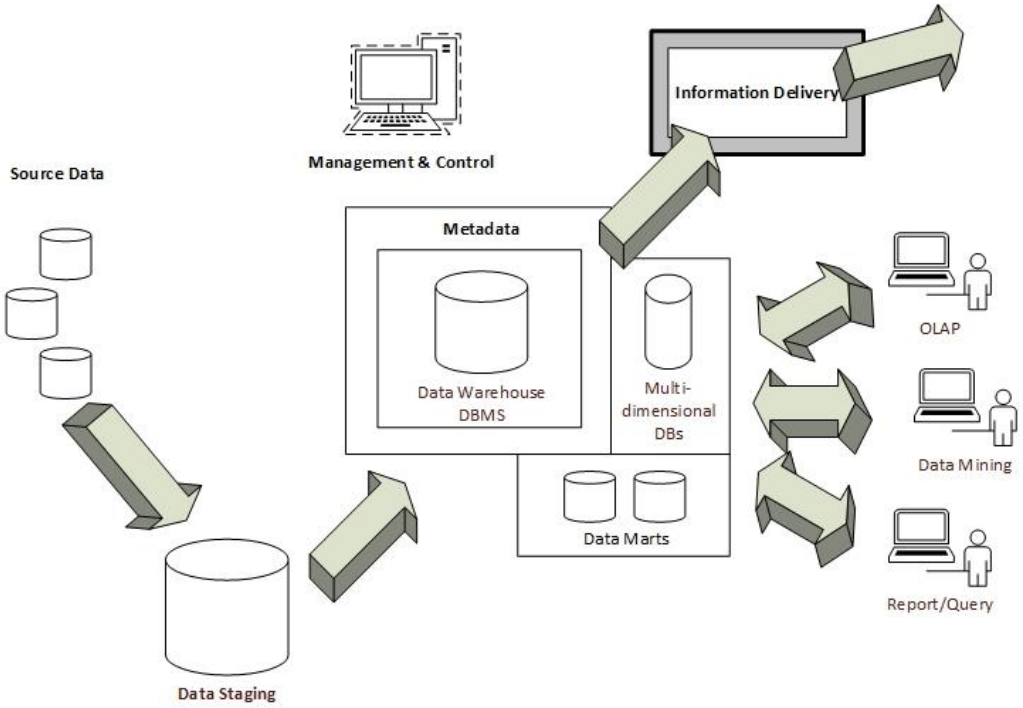
1. Berdasarkan model dimensional.
2. Mengandung *historical* data.
3. Terdiri dari *detailed* dan *summarized* data.
4. Tetap mempertahankan konsistensi data walaupun berasal dari sumber yang berbeda.
5. Fokus dalam *single subject*, seperti penjualan, keuangan, atau inventarisasi.

**Tabel 2.1 Perbedaan OLTP dan Data *Warehouse***

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

### 2.2.1.1 Arsitektur Data Warehouse

Arsitektur pada data *warehouse* dikelompokkan menjadi 5 bagian :




**Gambar 2.3 Arsitektur Data *Warehouse***

1. *Source* Data

   Merupakan sumber data, atau bisa dibilang bahwa darimana data itu berasal. Untuk membangun data *warehouse* di PT. Perkebunan Nusantara (PTPN) VIII Perkebunan Ciater maka source data berasal dari operasional system atau OLTP *database*.

2. Data *Staging*

   Merupakan sebuah proses yang diperlukan sebelum data source masuk ke dalam data *warehouse*. Proses ini dinamakan proses ETL (Extracting, Transformation, Loading).

3. Data *Warehouse*

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

4. Data *Marts*

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

5. *Users*

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

#### 2.2.1.2 Dimensional Data Warehouse

Kebutuhan *user* dan realitas data yang menjadi faktor penentu untuk merancang dimensional model data *warehouse*, seperti bisnis apa yang paling diperlukan, detailnya seperti apa dan dimensi-dimensi serta fakta-fakta apa yang harus diikutkan [4].

Maka dimensional model harus disesuai dengan kebutuhan dari *user*. Model juga harus dirancang sedemikian rupa agar dapat bertahan dan dapat beradaptasi dari segala perubahan yang akan terjadi. Desain modelnya yang dihasilkan dibentuk

menjadi *database* relasional yang mendukung OLAP *cubes* untuk menyediakan secara “instant” hasil *query* untuk analis.

1. Tabel Dimensi (*Dimension Tables*)

   Tabel dimensi menjelaskan tentang entitas bisnis dari suatu enterprise [4]. Tabel dimensi umumnya berisi data keterangan, dimana data tersebut jarang sekali mengalami perubahan.

2. Tabel Fakta (*Fact Tables*)

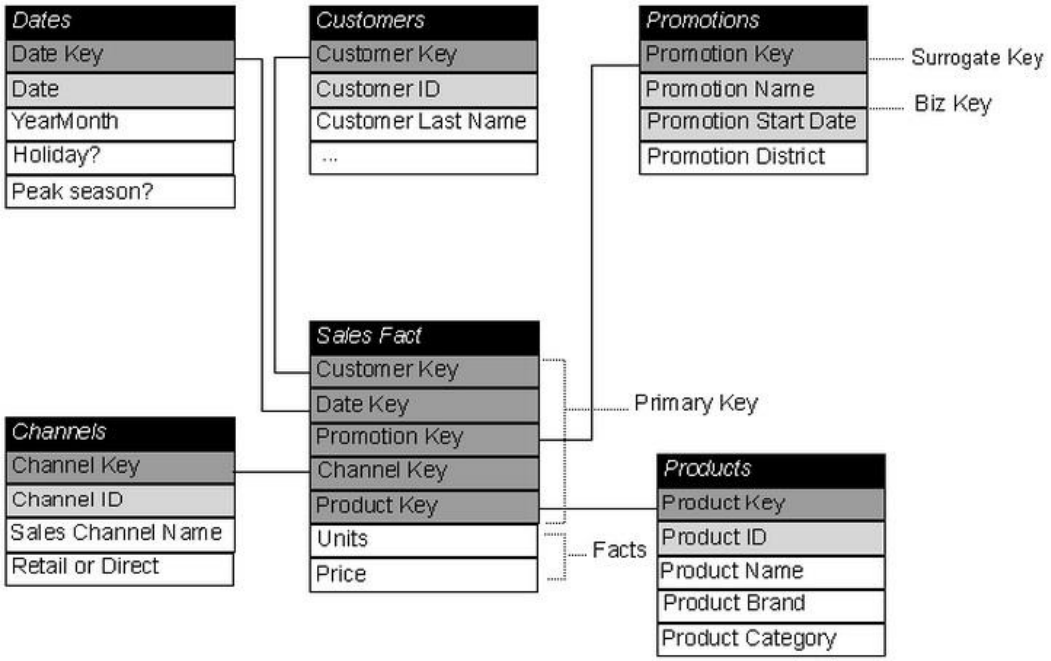
   Tabel fakta merupakan sebuah tabel yang menjelaskan tentang transaksi bisnis dari suatu enterprise biasanya disebut tabel detail [4]. Tabel fakta umumnya berisi data yang berkaitan langsung dengan proses bisnisnya.

3. Skema *Dimensional* Model

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

a) Skema *star*

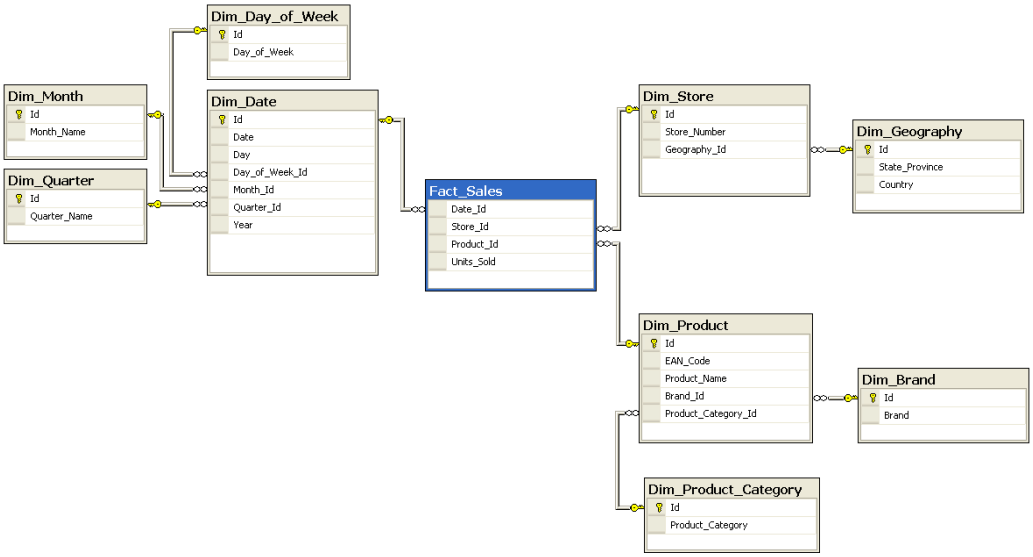
   Suatu skema disebut skema *star* jika seluruh tabel dimensi dihubungkan secara langsung ke tabel fakta dan satu tabel fakta wajib memiliki relasi minimal dengan satu tabel dimensi [4].




**Gambar 2.4 Skema Star**

b) Skema snowflake

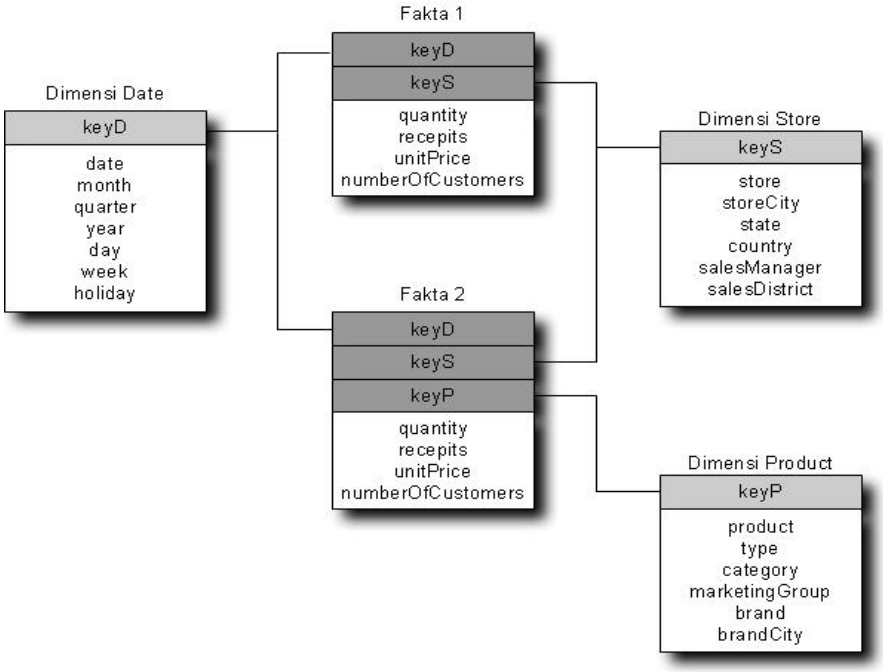
Suatu skema disebut skema *snowflake* jika satu atau lebih tabel dimensi tidak berhubungan langsung dengan tabel fakta tetapi harus berhubungan melalui tabel dimensi lain [4].




**Gambar 2.5 Skema Snowflake**

c) Skema constellation

Suatu skema dikatakan sebagai skema *constellation* jika ada satu tabel dimensi yang dipakai bersamaan oleh satu atau lebih tabel fakta [5].

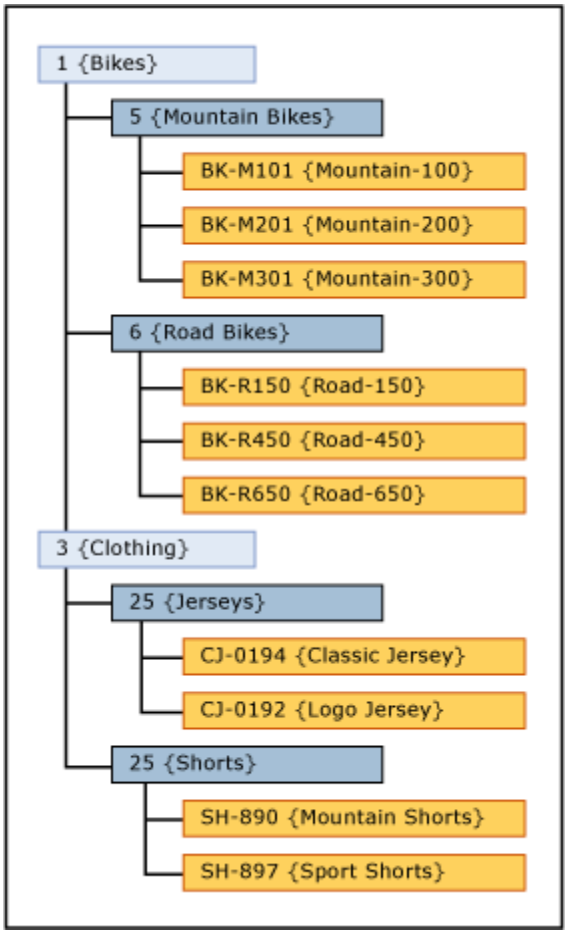



**Gambar 2.6 Skema *Constellation***

### 2.2.1.3 Hirarki (Hierarchies)

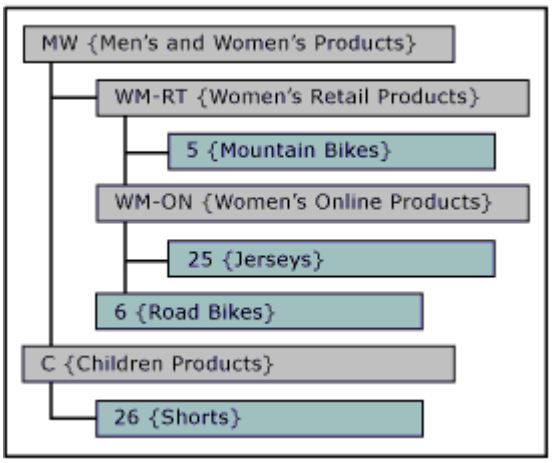
Hirarki merepresentasikan *relationships* antar tiap entitas ataupun tiap anggota entitasnya. Hirarki bisa juga dikatakan sebagai *parent-child relationships* karena hirarki memandang sesuatu melalui berbagai sudut pandang. Hirarki dibagi menjadi:

1. *Derived Hierarchies* merupakan suatu hirarki yang memiliki *relationships* antar entitas, sebagai contoh entitas kategori berhubungan dengan entitas produk, maka ada suatu cara untuk membangun *derived relationships* berdasarkan nilai dari atributnya. Contoh lainnya bisa dilihat sebagai berikut.




**Gambar 2.7 *Derived Hierarchies***

2. *Explicit Hierarchies*, merupakan suatu hirarki yang merepresentasikan *relationships* antar tiap member dalam sebuah entitas. Biar lebih jelas untuk *explicit hierarchies* bisa dilihat pada gambar dibawah ini.




**Gambar 2.8 *Explicit Hierarchies***

#### 2.2.1.4 Proses ETL Data Warehouse

*ETL* merupakan himpunan fungsi yang dilakukan untuk mengubah dan membentuk kembali data ke dalam bentuk yang berbeda pada data di dalam sistem operasional yang disimpan di dalam data *warehouse* sebagai informasi yang relevan dan strategis [4]. Adapun kelompok himpunan *ETL* adalah ekstraksi data, transformasi, dan *loading* yang menjadi tahapan proses pengubahan dan pembentukan ulang data yang digunakan di dalam data *warehouse* [4].

1. *Extraction*

   Tahap extraction merupakan tahap untuk mengidentifikasi semua sumber data internal, menentukan proses komputasi dan sumber data mana yang akan di ekstrak, menentukan kompatibilitas struktur data jika dan hanya jika bila menggunakan sumber-sumber dari luar, dan mengindikasi metode untuk mengekstraksi data [4].

2. *Transformation*

   Tahap *transform* memiliki fungsi meliputi pemilihan input, pemisahan struktur input, normalisasi dan denormalisasi dari struktur data sumber, mengagregasi, mengkonversi, dan memecahkan nilai yang hilang [4].

3. *Loading*

   Tahap *loading* merupakan tahap menginisialisasi *load* awal, menentukan seberapa sering suatu kelompok data harus tetap *up-to-date* dalam data *warehouse*, dan menentukan cara mengubah data yang akan dilaksanakan dalam periode waktu tertentu [4].

#### 2.2.1.5 OLAP (*On-Line Analytical Processing*)

OLAP merupakan salah satu kategori teknologi perangkat lunak yang memungkinkan analis, manajer, dan eksekutif untuk menggali suatu data secara cepat, konsisten, dan memiliki akses yang interaktif dalam luasnya penggalian suatu informasi yang di transformasikan dari suatu *raw* data menjadi dimensi fakta yang dapat dimengerti oleh pengguna [4].

1. Kriteria *OLAP*

Suatu sistem *OLAP* harus memiliki kriteria yang dapat menghasilkan suatu prioritas yang akan dituju. Adapun kriteria dari *OLAP* adalah sebagai berikut.

a. *Multidimensional Conceptual View*

Suatu model data *multidimensional* mangacu pada bagaimana peresepsi pengguna untuk memecahkan suatu masalah bisnis yang dapat menciptakan suatu analisis yang intuitif dan mudah digunakan [4].

b. *Transparency*

Teknologi, gudang data, arsitektur komputasi, dan data alami dari sumber ditransparasikan terhadap pengguna, dengan tujuan untuk dapat membantu meningkatkan keefisienan dan produktivitas pengguna [4].

c. *Accessibility*

Suatu akses yang dihasilkan untuk menampilkan data harus dapat menghasilkan hasil yang dapat di analis secara spesifik, dipresentasikan sebagai tunggal, konherensi, dan konsisten bagi pengguna yang melihatnya [4].

d. *Consistent Reporting Performance*

*OLAP* memastikan kekonsistenan penggunaan dimensi atau ukuran yang digunakan sehingga pengguna dapat melaporkan kinerja dengan efisien terhadap waktu baik itu waktu dijalankan, waktu respon, atau pemanfaat sistem setiap kali *query* digunakan [4].

e. *Client/Server Architecture*

Pembangunan *OLAP* sesuai dengan prinsip arsitektur klien atau server agar kinerja optimal, fleksibel, mampu beradaptasi, dan inter-operabilitas [4]. Sehingga pengguna tidak perlu menginputkan *query* lagi dalam pembentukan ulang [4].

f. *Generic Dimensionality*

*OLAP* memastikan terhadap setiap dimensi memiliki kesetaraan dalam struktur, kemampuan operasional, dan memiliki satu struktur yang logis [4].

g. *Multi User Support*

*OLAP* memberikan dukungan pada *end user* untuk bekerja secara bersamaan dengan model analisis yang sama atau untuk model yang berbeda pada data yang sama [4].

h. *Intuitive Data Manipulation*

*OLAP* memungkinkan berbagai macam jalur konsolidasi re-orientasi pada pemanipulasian informasi seperti *drill-down, roll-up,* dan manipulasi yang dilakukan secara intuitif dan langsung [4].

i. *Flexible Reporting*

*OLAP* memberikan kebebasan pengguna untuk mengatur dengan mudah suatu kolom, baris dan sel dengan fasilitas manipulasi yang mudah, analisis, dan sintesis informasi [4].

j. *Unlimited Dimensions and Agregation Levels.*

*OLAP* dapat mengakomodasi banyak dimensi data dalam suatu model umum analisis [4].

2. Karakteristik *OLAP*

Karakteristik dasar dari *OLAP* adalah sebagai berikut.

a. Memastikan pengguna memiliki suatu *view* yang *multidimensional* dan *logical* dalam suatu data warehouse [4].
b. Memfasilitasi *query* yang interaktif dan analisis yang kompleks untuk pengguna [4].
c. Memungkinkan user untuk melakukan *drill down* untuk mendapatkan detail yang lebih rinci atau melakukan *roll up* untuk proses agregasi dari suatu *metrics* dalam suatu dimensi tunggal ataupun dalam *multiple* dimensi [4].
d. Menghasilkan suatu *view* yang dapat menyajikan arti dari segala arah, termasuk *chart* dan *graph* [4].

### 2.2.2 Microsoft SQL Server

Microsoft SQL server adalah sebuah sistem manajemen basis data relasional (RDBMS) produk microsoft. Bahasa kueri utamanya adalah transast-sql yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh microsfot. Umumnya SQL server digunakan di dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudia berkembang dengan digunakannya SQL server pada basis data besar [8].

Microsoft SQL server dan sybase/ASE dapat berkomunikasi lewat jaringan dengan menggunakan protokol TDS (tabular Data Stream). Selain dari itu, SQL server juga mendukung ODBC (Open Database Connectivity), dan mempunyai driver JDBC untuk bahasa pemrograman java. Fitur yang laind ari SQL server ini adalah kemampuannya untuk membuat basis data mirroring dan clustering. Pada tahun 2000 Microsoft mengeluarkan SQL Server 2000 yang merupakan versi yang banyak digunakan. Berikut ini adalah beberapa fitur yang dari sekian banyak fitur yang ada pada SQL Server 2000 [8]:

a) *XML Support.* Dengan fitur ini, Anda bisa menyimpan dokumen XML dalam suatu tabel, meng-*query* data ke dalam format XML melalui *Transact*-SQL dan lain sebagainya [8].
b) *Multi-Instance Support.* Fitur ini memungkinkan Anda untuk menjalankan beberapa *database engine* SQL Server pada mesin yang sama [8].
c) *Data Warehousing and Business Intelligence (BI) Improvements.* SQL Server dilengkapi dengan fungsi-fungsi untuk keperluan *Business Intelligence* melalui *Analysis Services*. Selain itu, SQL Server 2000 juga ditambahi dengan tools untuk keperluan data mining [8].
d) *Performance and Scalability Improvements.* SQL Server menerapkan *distributed partitioned views* yang memungkinkan untuk membagi

*workload* ke beberapa server sekaligus. Peningkatan lainnya juga dicapai di sisi DBCC, *indexed view*, dan *index reorganization* [8].

e) *Query Analyzer Improvements.* Fitur yang dihadirkan antara lain: *integrated debugger, object browser,* dan fasilitas *object search* [8].

### 2.2.3 Object Oriented Analysis and Design (OOAD)

OOAD merupakan teknik dan metode dalam mengembangkan sistem yang berbasis pada objek, di mana OOAD menggabungkan data dan proses menjadi suatu entitas tunggal yang disebut objek. Tujuan dari OOAD adalah membuat elemen sistem yang lebih *reusable, improving system quality,* dan produktivitas terhadap analisis dan desain sistem. *Tools* yang digunakan dalam melakukan pemodelan pada OOAD adalah UML ( *Unified Modeling Language ).* Berikut adalah diagram yang digunakan dalam UML [10].

1. *Use Case*

   *Use Case* merupakan situasi yang digunakan oleh sistem untuk memenuhi satu atau dua kebutuhan dari pengguna. *Use Case* menggambarkan sistem berdasarkan hasil dari *requirement gathering,* setiap *use case* melambangkan untuk satu fungsional sistem [10].

2. *Activity Diagram*

   *Activity Diagram* memungkinkan bagaimana sistem mencapai tujuan dalam setiap tindakan yang dilakukan pada suatu proses yang telah digambarkan pada *use case* sebelumnya [10].

3. *Class Diagram*

   *Class Diagram* menjelaskan hubungan dari *class* yang digunakan pada sistem. *Class Diagram* menggambarkan jenis objek beserta relasinya yang ada pada sistem [10].

4. *Sequence Diagram*

   *Sequence Diagram* merupakan sebuah gambaran interaksi pada sistem yang didasarkan ketika suatu proses *use case* telah dijalankan pada sistem tersebut [10].