

LECTURE NOTES

7023T Advanced Database System

Session 06

Dimensional Modelling 2

LEARNING OUTCOMES

- Peserta diharapkan mampu memahami konsep *dimensional modelling*
- Peserta diharapkan dapat menjelaskan karakteristik tabel fakta dan tabel dimensi.
- Peserta diharapkan mampu mengidentifikasi berbagai jenis tabel fakta dan tabel dimensi

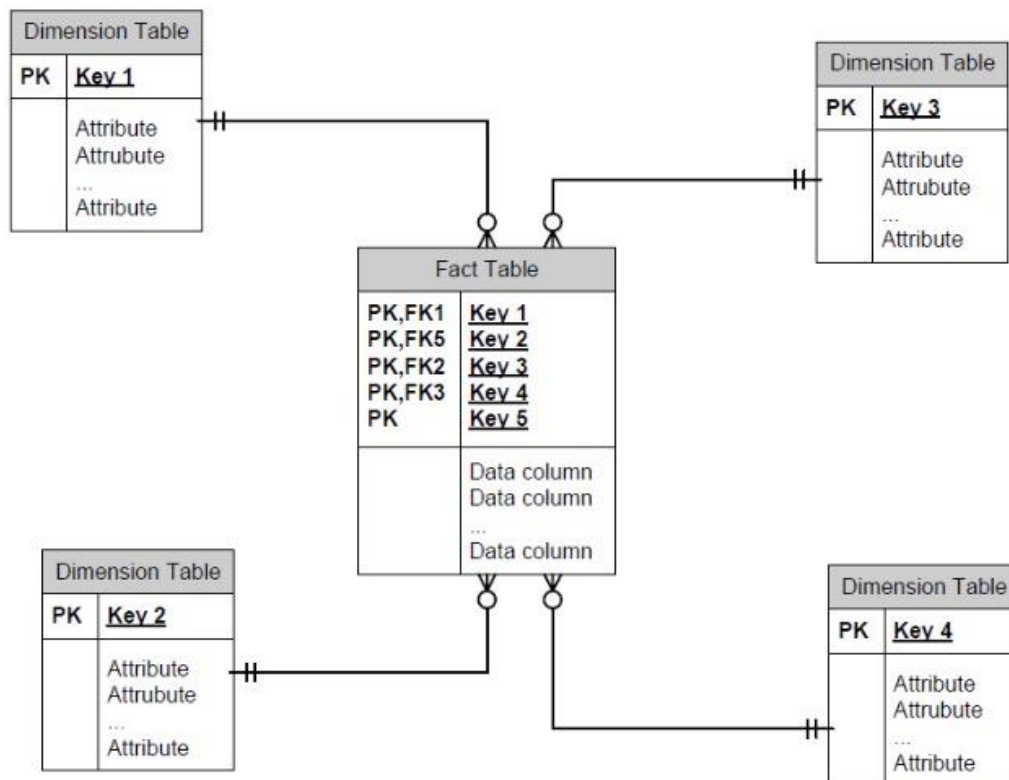
OUTLINE MATERI (Sub-Topic):

1. *Dimensional Models*
2. Tabel Fakta
3. Tabel Dimensi

Dimensional Models

Dimensional model adalah model data yang dirancang agar mudah dipahami dan mudah diakses (melalui *query*) oleh pengguna dari kalangan bisnis, sekaligus mendapatkan respon *query* yang cepat. Model ini mudah dipahami karena sudah dilakukan pengelompokkan terhadap kategori bisnis. Pengguna dari kalangan bisnis sehingga dapat melakukan navigasi pada model dengan mudah dan mengabaikan kategori yang tidak relevan dengan kebutuhannya. Model akan menghasilkan respon *query* yang cepat dengan cara meminimalkan jumlah join antar tabel untuk menjawab permintaan dari pengguna dari kalangan bisnis.

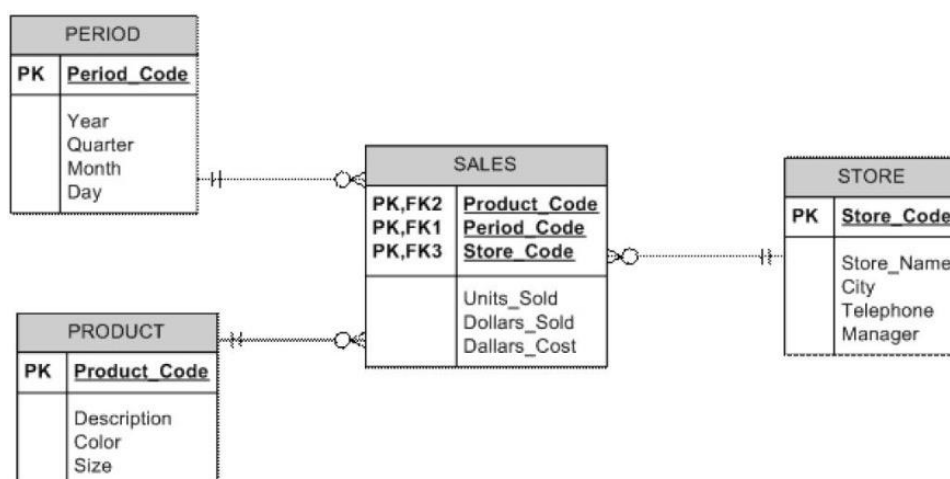
Dimensional models disimpan dalam database relasional dalam struktur *star schema*. Untuk keperluan OLAP analysis maka *dimensional models* disimpan dalam struktur multidimensi yang disebut sebagai OLAP. Sebagai catatan, baik *dimensional models* maupun model 3NF adalah model logikal, sehingga keduanya dapat disimpan secara fisik dalam database relasional. Kedua model dapat direpresentasikan dengan *entity-relationship diagram* (ERD) seperti diilustrasikan pada gambar 1.



Gambar 1. Contoh *star schema* direpresentasikan dalam ERD

Dimensional model seringkali disebut sebagai *denormalized models*, dimana tabel fakta dinormalisasikan menjadi model 3NF sedangkan tabel dimensi biasanya dimodelkan dengan 2NF. Tingkat denormalisasi ini menghasilkan model data yang lebih mudah dipahami dan diakses dengan *query* oleh pengguna dari kalangan non-IT. Sebaliknya, model 3NF dirancang untuk mengeliminasi redundansi data dan mengoptimalkan transaksi yang akan melakukan operasi *update* pada tabel. Karena pengguna dari kalangan bisnis lebih banyak melakukan akses data berupa operasi pemilihan data (*select query*) dibanding operasi *update*, maka model 3NF bukanlah pilihan terbaik bagi mereka.

Dimenasional modelling adalah metode untuk merancang *dimensional model*. *Dimensional modelling* akan membagi data menjadi fakta (pengukuran-pengukuran) dan dimensi (konteks). Fakta pada umumnya berupa data numerik yang diperoleh dari data operasional yang mendukung proses bisnis dari suatu organisasi. Dimensi biasanya berupa data tekstual yang merepresentasikan konteks dimana fakta diukur dan direkam pada database. Dimensi menjelaskan siapa, kapan, dimana, mengapa, dan bagaimana fakta direkam. Setiap proses bisnis akan direpresentasikan oleh sebuah tabel fakta, yang mengandung fakta-fakta, dan sekumpulan tabel dimensi yang menyimpan dimensi. Gambar 2 memperlihatkan contoh dari tabel fakta (SALES) dan tiga tabel dimensi (PERIOD, PRODUCT, dan STORE).

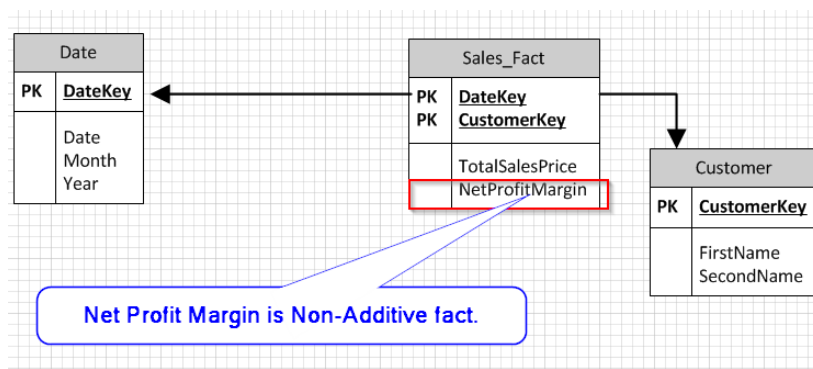
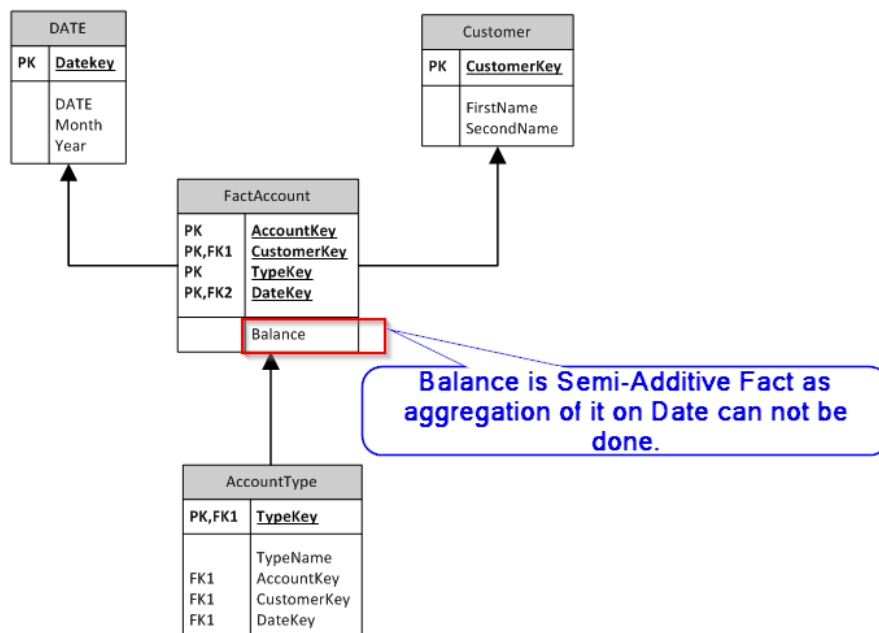
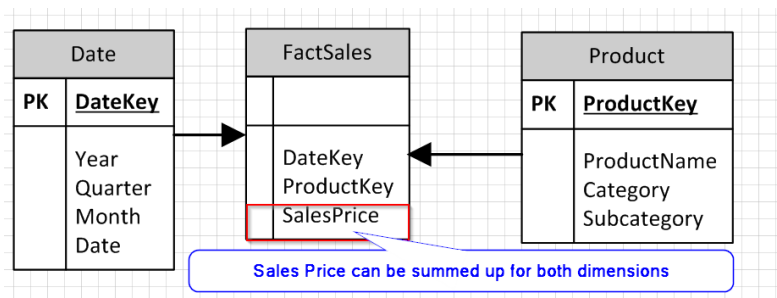


Gambar 2. Contoh sebuah tabel fakta dengan tiga tabel dimensi

Tabel Fakta

Tabel fakta menyimpan data numerik yang merupakan nilai variabel (nilainya belum diketahui sebelum dilakukan pengukuran). Nilai numerik yang merupakan konstanta biasanya

merupakan dimensi, namun dalam beberapa kasus dapat juga sebuah konstanta merupakan dimensi juga fakta. Jenis fakta yang paling berguna adalah *additive*, dimana nilainya dapat dijumlahkan terhadap semua dimensi yang berasosiasi dengan tabel fakta. Jenis fakta lain adalah *semi-additive*, dimana nilainya tidak dapat dijumlahkan terhadap sebagian dimensi. Sedangkan fakta *non-additive* nilainya tidak dapat dijumlahkan terhadap dimensi apapun. Gambar 3 memperlihatkan fakta *non-additive*, *semi-additive*, dan *non-additive*.



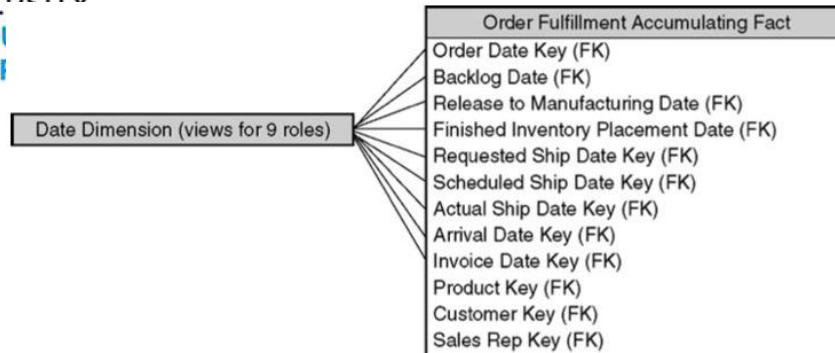
Gambar 3. Contoh fakta *additive*, *semi-additive*, dan *non-additive*

Tabel fakta menyimpan lebih dari 90% dari keseluruhan data, sehingga ukuran dari tabel ini sangat besar bahkan bisa mengandung jutaan baris data. Untuk alasan efisiensi, maka setiap fakta disimpan pada satu tabel fakta yang dapat diakses oleh semua unit dalam suatu organisasi. Tabel fakta harus memiliki beberapa *foreign key* yang berfungsi untuk menghubungkan tabel fakta ke semua tabel dimensi. *Foreign key* dari tabel fakta tidak boleh bernilai null, jika tidak maka relasi ke tabel dimensi tidak dapat dibuat.

Tabel fakta disarankan memiliki tingkat *granularity* (*grain*) yang paling detail sehingga memungkinkan untuk menjawab pernyataan yang lebih detail dari pengguna. Hasil pengukuran yang disimpan pada tabel fakta dapat berupa transaksional, *periodic snapshot*, atau *accumulating snapshot*. Tabel fakta transaksional menyimpan baris baru setiap ada transaksi baru. Pada tabel fakta *periodic snapshot*, setiap baris baru akan ditambahkan secara periodik (misal setiap bulan), contohnya adalah laporan keuangan yang selalu diperbaharui setiap bulannya. Tabel fakta *accumulating snapshot* akan menyimpan baris baru untuk setiap *snapshot*, namun akan melakukan pembaharuan (*update*) terhadap atribut yang menjelaskan fase transaksi yang dimaksud. Gambar 4 memperlihatkan contoh dari tabel fakta *periodic snapshot* yang menjelaskan transaksi pembayaran angsuran bulanan kendaraan bermotor, sedangkan gambar 5 memperlihatkan contoh dari tabel fakta *accumulating snapshot*, yang menjelaskan proses pemenuhan order yang dimulai dari beberapa fase seperti: *order*, *backlog*, *release to manufacturing*, *finished inventory placement*, *requested shipment*, *scheduled shipment*, *actual shipment*, dan *arrival*. Informasi yang dicatat pada setiap fase tersebut adalah tanggal.



Gambar 4. Contoh tabel fakta *periodic snapshot*



Gambar 5. Contoh tabel fakta *accumulating snapshot*

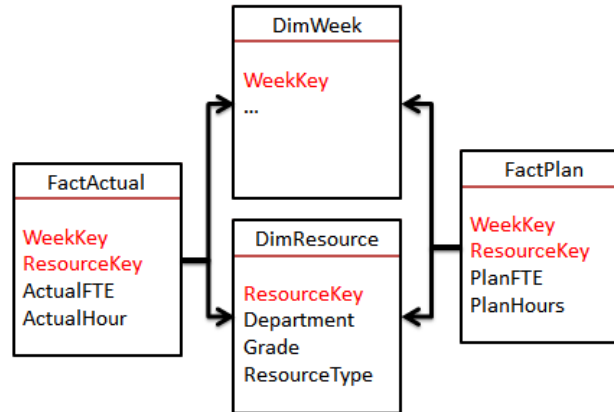
Tabel Dimensi

Tidak seperti tabel fakta yang tersusun atas key dan hasil pengukuran numerik, tabel dimensi tersusun atas atribut yang bersifat deskriptif. Atribut-atribut tersebut digunakan untuk *constraint/filter* pada *query* dan *report*. Dimensi biasanya berkorespondensi dengan kata “berdasarkan” (*by*), contoh pada gambar 1 setiap dimensi merepresentasikan fakta berdasarkan waktu (PERIOD), produk (PRODUCT), dan outlet (STORE). Atribut tabel dimensi biasanya bersifat deskriptif dan lengkap (tidak boleh mengandung *missing values*). Kode atau singkatan dapat digunakan pada tabel dimensi untuk menyingkat representasi data.

Tabel dimensi biasanya merepresentasikan relasi *many-to-one*, hal ini merupakan konsekuensi dari denormalisasi. Contoh, sebuah produk dijual pada milih toko tertentu, sedangkan toko terletak pada provinsi tertentu, maka kedua hirarki tersebut direpresentasikan dengan sebuah dimensi untuk produk yang memiliki atribut toko dan provinsi. Membuat dimensi terpisah untuk toko dan provinsi akan mengurangi kebutuhan ruang penyimpanan, namun model yang dihasilkan akan lebih sulit dipahami dan menurunkan kinerja *query*.

Dua tabel dimensi disebut *conformed* jika keduanya identik untuk semua tabel fakta yang berelasi dengannya. Dua dimensi juga dikatakan *conformed* jika salah satu darinya (dengan tingkat granularity lebih rendah) adalah subset dari yang lain. *Standardized conformed dimensions* (atau *master dimensions*) merupakan hal penting dalam data warehouse karena akan menjamin konsistensi dan mempermudah integrasi dari beberapa proses bisnis. Gambar 6 memperlihatkan contoh *conformed dimension*. Dimensi DimWeek dan Dim Resource pada gambar tersebut adalah *conformed dimension* karena kedua tabel tersebut berelasi dengan dua tabel fakta berbeda, yaitu FactActual dan FactPlan. Tabel-tabel tersebut disebut sebagai “*bus*” dari enterprise data warehouse, karena mereka

menghubungkan beberapa tabel fakta yang masing-masing merepresentasikan proses bisnis yang berbeda. Penggunaan *standardized conformed dimension* perlu dioptimalkan agar diperoleh kinerja terbaik dari enterprise data warehouse.



Gambar 6. Contoh *conformed dimension*

Semua tabel dimensi untuk sebuah proses bisnis dihubungkan satu sama lain dengan tabel fakta yang sama, sehingga membentuk struktur yang serupa dengan bentuk bintang, oleh karena itu dikenal sebagai *star schema*. Sifat simetrik dari struktur ini memungkinkan *database engine* mengeksekusi *query* secara optimal dan terprediksi. Tabel memiliki hanya satu atribut yang merupakan *key*. Disarankan untuk memanfaatkan *integer surrogate key* daripada *primary key* dari tabel sumber, hal ini akan meningkatkan kinerja *query*.

SIMPULAN

- Hasil proses *dimensional modelling* adalah *denormalized* model yang dioptimalkan untuk kemudahan pemahaman dan kinerja *query*.
- Tabel fakta merepresentasikan transaksi yang terjadi pada masing-masing proses bisnis dengan konteks yang direpresentasikan oleh masing-masing tabel dimensi.
- *Standardized conformed dimensions* akan mengoptimalkan kinerja enterprise data warehouse karena sebuah tabel dimensi dapat digunakan oleh lebih dari satu proses bisnis atau tabel fakta.

DAFTAR PUSTAKA

1. Kimball, R. (2008). *The Data Warehouse Lifecycle Toolkit*. John Wiley & Sons.
2. Kimball, R., & Ross, M. (2011). *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. John Wiley & Sons.
3. Inmon, W. H. (2005). *Building the Data Warehouse*. John wiley & sons.