Data Warehouse (II): Pemodelan Berdimensi (Lanj.)

Yudi Agusta, PhD
Data Warehouse and Data Mining, Lecture 4

Copyright © Yudi Agusta, PhD 2006

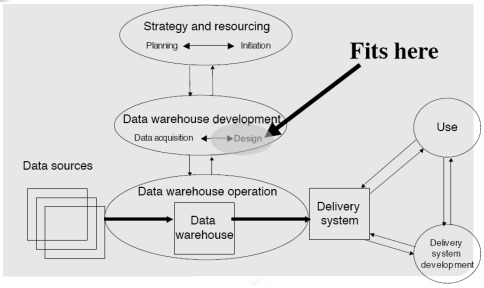
Lecture's Structure

- Rekap Kuliah Lalu
- Contoh Rancangan
  - Inventory
  - Layanan Keuangan
- Rancangan Berdimensi
- Agregat

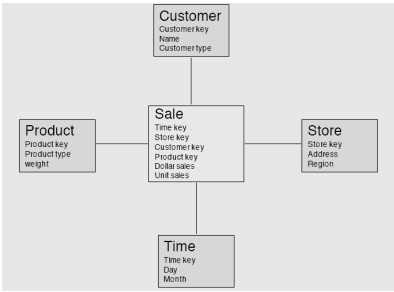
Rekap

- Pemodelan Berdimensi
  - Populer, Berguna dan pendekatan pragmatis
  - Berdasarkan teori Kimball
- Tabel fakta
- Tabel Berdimensi
- Proses Perancangan Menggunakan Tahapan

Skema Rancangan Database



Skema Star (Dengan Atribut)



Tahapan Dalam Proses Perancangan

- Memilih proses bisnis
- Memilih inti dari tabel fakta
- Memilih dimensi
- Memilih fakta yang terukur (umumnya numeric, additive quantities)
- Melengkapi tabel berdimensi
  - (Kimball, 1996)

Tahapan Tambahan dalam Proses Perancangan

- Menentukan strategi untuk secara perlahan mengganti dimensi
- Membuat agregat dan komponen penyimpanan fisik lainnya
- Menentukan jangka waktu historical dari database
- Menentukan tingkat keperluan data yang mana yang perlu untuk diekstrak dan mana yang perlu dimasukkan ke dalam data warehouse

Contoh Rancangan

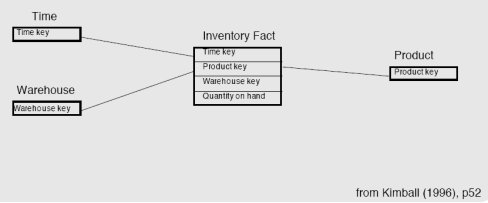
- Cara bagus untuk belajar mengenai prinsip perancangan data warehouse adalah dengan menggunakan contoh
  - Penggunaan ulang
- Kimball – Data Warehouse Lifecycle Toolkit
- Adamson & Venerable – Data Warehouse Design Solutions
- Kita akan melihat permasalahan inventory, pengiriman dan layanan keuangan

Inventory

- Sistem inventory memberikan servis sebagai 'penengah' antara perusahaan manufaktur dan perusahaan retail
  - Proses penambahan nilai
- Ada tiga jenis model inventory
  - Potret Inventory
  - Status Pengiriman
  - Transaksi

Model Potret Inventory

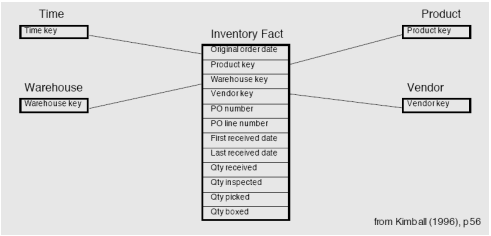
- Untuk jangka waktu tertentu, level inventory diukur dan dicatat



from Kimball (1996), p52

Model Status Pengiriman

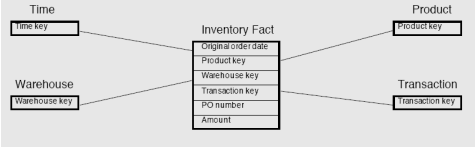
- Membuat satu catatan untuk setiap pengiriman suatu produk ke dalam warehouse



from Kimball (1996), p56

Model Transaksi

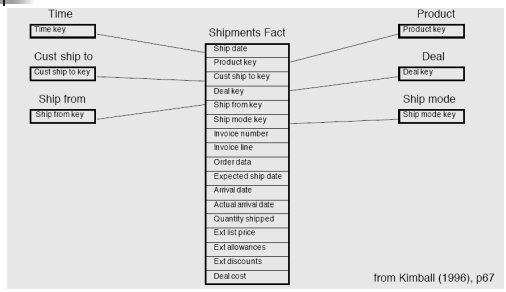
- Mencatat setiap transaksi yang mempengaruhi inventory



Pengiriman

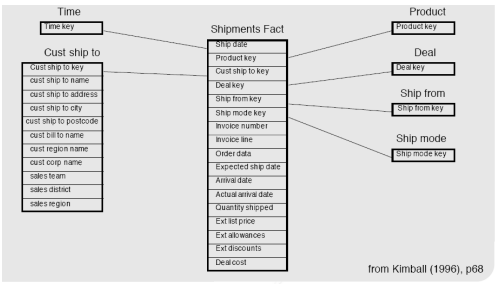
- Proses pengiriman adalah dimana produk meninggalkan sebuah perusahaan dan diterima oleh perusahaan lainnya
- Umumnya, yang ikut dalam suatu pengiriman adalah invoice pengiriman

Pengiriman



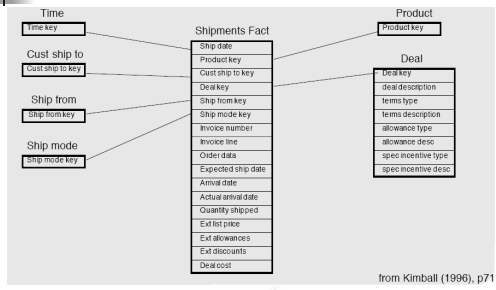
from Kimball (1996), p67

Pengiriman



from Kimball (1996), p68

Pengiriman



from Kimball (1996), p71

Layanan Keuangan

- Umumnya bank besar
- Layanan mencakup:
  - Cheque account, savings account, mortgage loans, investment loans, credit cards etc.
- Goal
  - Untuk memasarkan produk ke setiap keluarga secara efektif
- Membangun data warehouse keluarga untuk melacak accounts, pemilik account, dan pengelompokan keluarga tersebut

Layanan Keuangan

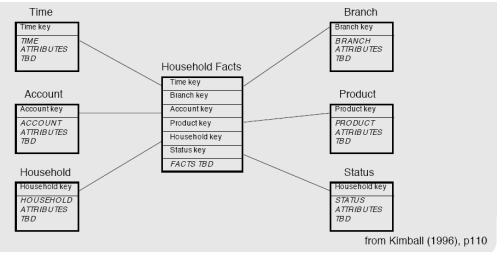
- Yang Diperlukan
  - Lima tahun dari data bulanan untuk setiap account
  - Untuk bulan sekarang harus diprotret dari hari sebelumnya
  - Setiap tipe account mempunyai atribut dan fakta numeric yang berbeda-beda
  - Setiap account terkait dengan suatu keluarga
  - Catatan nama dan alamat mengenai pemilik account mungkin berbeda untuk setiap account
  - Menarik untuk mencari penyebaran dan aktivitas dari setiap account

Layanan Keuangan

- Pilih proses bisnis
  - Saldo account per bulan
- Pilih inti dari tabel fakta
  - Saldo untuk setiap account berdasarkan bulan

Layanan Keuangan

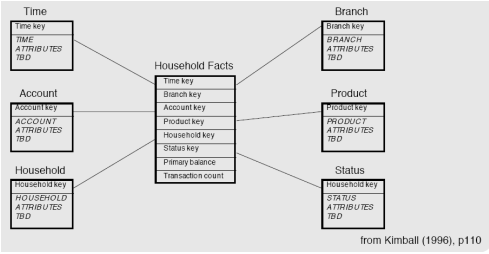
- Memilih dimensi



from Kimball (1996), p110

Layanan Keuangan

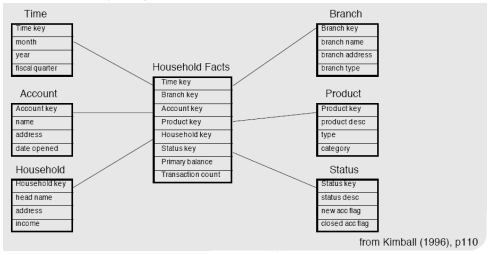
- Memilih fakta terukur



from Kimball (1996), p110

Layanan Keuangan

- Melengkapi Tabel Dimensi



from Kimball (1996), p110

Produk Dengan Jenis Berbeda

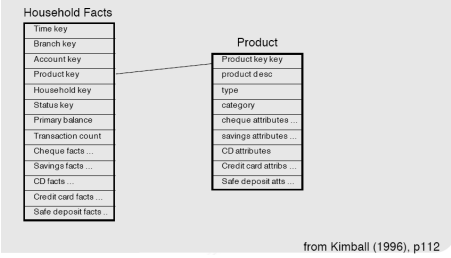
- Merupakan situasi umum di perbankan
- Setiap tipe account mempunyai sejumlah fakta yang tidak terasosiasikan dengan tipe account
  - Savings
    - Jumlah bunga yang dibayarkan
  - Cheque
    - Overdraft limit
  - Credit cards
    - Credit limit

Secara Logic – Satu Tabel Fakta

- Dalam keadaan seperti ini, rancangan logic dari tabel fakta mencakup semua fakta tambahan dan atribut dimensi di dalam tabel
- Dapat dengan mudah mencakup lusinan atau lebih atribut untuk setiap jenis account atau produk

Layanan Keuangan

- Produk Berbagai Jenis

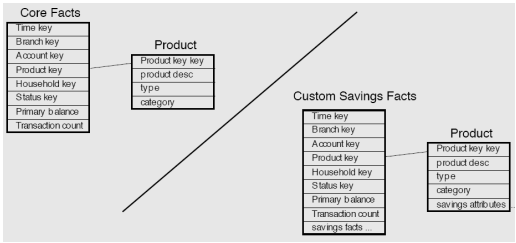


from Kimball (1996), p112

Secara Riil - Bencana

- Setiap fakta dan tabel produk dapat mempunyai lebih dari 100 field
  - Terlalu mahal bila dihitung dari segi media penyimpan, kecepatan dan tingkat kekompleksan
- Kebanyakan fields dalam sebagian besar records akan kosong
- Jawaban Kimball adalah untuk memecah fakta dan tabel dimensi sesuai dengan tipe produk
- Pekerjaan tambahan bagi meta data!

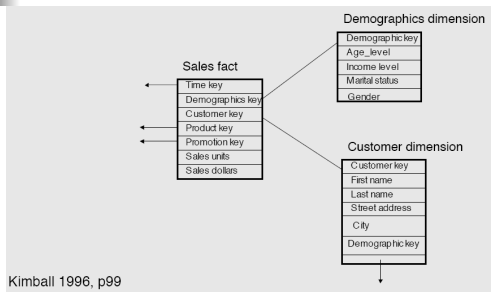
Layanan Keuangan: Fakta Utama dan Yang Dibuat



Dimensi-Mini

- Umumnya, digunakan untuk melihat sebaran atau kategori
- Membuat link antara tabel dimensi
- Digunakan untuk mengelompokkan nilai yang bersifat continuous ke dalam grup
  - Contoh: Pendapatan, Umur dll
- Mungkin mempunyai lebih dari 1, 2-3 bukan tidak biasa

Dimensi-Mini Sebaran Umum



Kimball 1996, p99

Tahapan dalam Proses Perancangan

- Memilih proses bisnis
- Memilih inti dari tabel fakta
- Memilih dimensi
- Memilih fakta terukur (umumnya numeric, additive quantities)
- Melengkapi tabel dimensi
  - Kimball 1996

Tahapan tambahan dalam proses perancangan

- Menentukan strategi untuk mengubah dimensi secara perlahan
- Membuat agregat dan komponen penyimpan riil lainnya
- Menentukan jangka waktu historical dari database
- Menentukan tingkat keperluan data yang mana yang perlu untuk diekstrak dan menyimpannya ke dalam data warehouse

Dimensi Yang Berubah Secara Perlahan

- Banyak dimensi (seperti Produk dan Pelanggan) berkembang secara perlahan dalam jangka waktu yang panjang
  - Manusia mengubah nama, alamat dll
  - Tim penjualan mengubah nama wilayah dll
- Tiga standar pendekatan a.l.:
  - Timpa nilai yang lama
  - Buat record dimensi tambahan
  - Buat field dengan nilai saat ini

Tipe 1: Timpa

- Timpa nilai atribute
  - Contoh: perubahan status perkawinan dari pelanggan bernama Mary Jones menjadi "married"
- Keuntungan: mudah untuk diimplementasikan
- Kekurangan: tidak ada catatan perubahan

Tipe 2: Record Baru

- Membuat satu tabel dimensi baru untuk setiap versi
  - Perlu untuk mengeneralisasikan kunci dimensi (menambahkan 2 atau 3 digit untuk range yang mencukupi)
- Keuntungan: Secara otomatis memelihara dan membagi catatan
- Kerugian: Lebih kompleks dari proses Timpa

Tipe 3: Membuat Field dengan Nilai Saat Ini

- Buat suatu field yang disebut dengan "Current X"
  - Current Address
  - Old Address
- Berguna bila kita ingin tahu nilai yang lama dan nilai yang baru
  - Contoh: Penyesuaian tim penjualan
- Keuntungan: Sederhana dan Cepat, mengijinkan untuk melakukan perbandingan
- Kerugian: Bagaimana dengan perubahan yang lainnya seperti perubahan tanggal dll

Memilih Metode

- Tipe 1
  - Sangat sederhana. Hanya digunakan bila nilai histori tidak begitu penting
- Tipe 2
  - Secara umum, sangat fleksibel. Kimball mengatakan, ini berguna untuk dimensi sampai ratusan dengan record berjumlah ribuan, khususnya disain dengan DBMS yang bagus
- Tipe 3
  - Tidak sefleksibel Tipe 3, tetapi dapat digunakan untuk dimensi dengan record yang berjumlah jutaan

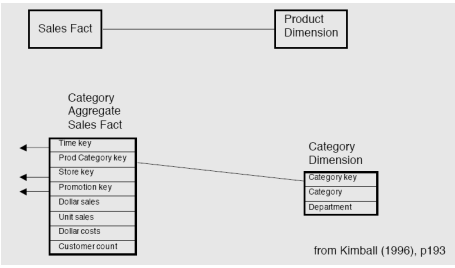
Agregat

- Sebagian besar data warehouse mempunyai tabel fakta dalam jumlah yang sangat besar (sampai 50 triliun record dan memerlukan media penyimpan sampai 1 – 5 terabytes)
- Agregat (summary sebelum disimpan) adalah cara yang paling efektif untuk meningkatkan performance dari data warehouse

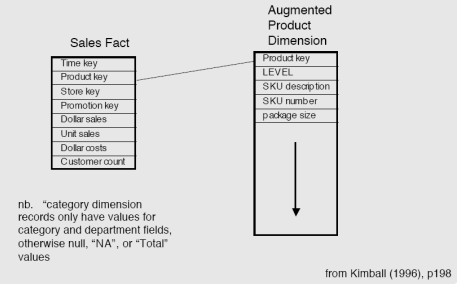
Agregat Penyimpan

- Agregat record dari tabel fakta yang merepresentasikan summary dari record tabel fakta pada level dasar
- Fakta agregat terletak pada tabel fakta agregat baru atau tabel fakta original
  - Yang maa yang terbaik?

Tabel Fakta Agregat



Field Level Baru

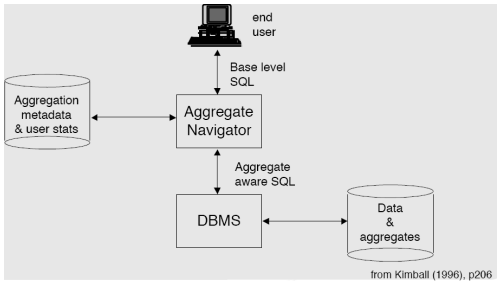


from Kimball (1996), p198

Agregat Penyimpan

- Field Level dapat menciptakan double counting pada saat melakukan queries
- Setiap inti dari agregat harus tersimpan di dalam tabel fakta sendiri, dan didukung dengan tabel dimensi kategori yang sesuai
  - Apa yang dipengaruhi terhadap jumlah tabel?
  - Bagaimana kekompleksan dari sudut pandang pengguna?

Agregat Pemandu



from Kimball (1996), p206

Agregat Pemandu

- Agregat pemandu secara otomatis mentransformasikan SQL berbasis pengguna ke SQL yang memperhatikan agregat
- Agregat pemandu secara dinamis memilih tabel agregat terbaik untuk digunakan
- Agregat pemandu mengisolasi pengguna dari agregat portfolio dan mengijinkan DBA untuk melakukan adjustment terhadap agregat