

KONSOLIDASI DATA WAREHOUSE UNTUK APLIKASI BUSINESS INTELLIGENCE

Rudy

Information Systems Department, School of Information Systems, Binus University
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480
rudy@binus.edu; rudy2105@yahoo.com

ABSTRACT

As the business competition is getting strong, corporate leaders need complete data that as a basis for determining future business strategies. Similarly with management of company "A", a pharmaceutical company which has three distribution companies. Each distribution company already has a data warehouse to generate reports for each of them. For business operational and corporate strategies, chairman PT "A" requires an integrated report, so analysis of data owned by the three distribution companies can be done in a full report to answer the problems faced by the management. Thus, data warehouse consolidation can be used as a solution for company "A". Methodology starts with analysis of information needs to be displayed on the application of business intelligence, data warehouse consolidation, ETL (extract, transform and load), data warehousing, OLAP and Dashboard. Using data warehouse consolidation, information access by management of company "A" can be done in a single presentation, which can display data comparison between the three distribution companies.

Keywords: data warehouse, consolidation, business intelligence

ABSTRAK

Dengan persaingan bisnis sangat ketat, pimpinan perusahaan membutuhkan data lengkap yang dapat dijadikan sebagai dasar menentukan strategi bisnis kedepan. Demikian pula dengan pimpinan PT "A" yang bergerak dalam bidang farmasi dan memiliki tiga perusahaan distribusi. Masing-masing perusahaan distribusi sudah memiliki data warehouse yang menghasilkan laporan untuk masing-masing perusahaan distribusi. Dalam menjalankan bisnis dan menentukan strategi perusahaan, pimpinan PT "A" membutuhkan laporan yang terintegrasi, sehingga dapat melakukan analisis data-data yang dimiliki ketiga perusahaan distribusi dalam sebuah laporan yang lengkap. Untuk menjawab permasalahan yang dihadapi di atas, solusi konsolidasi datawarehouse dapat diterapkan pada PT "A". Metodologi yang digunakan adalah melakukan analisis kebutuhan informasi dalam hal ini informasi yang akan ditampilkan pada aplikasi business intelligence dan melakukan konsolidasi data warehouse dengan arsitektur sumber data, ETL (extract, transform and load), data warehouse, OLAP dan Dashboard. Dengan adanya konsolidasi data warehouse, akses informasi oleh pimpinan perusahaan dapat dilakukan dalam sebuah layar presentasi dengan menampilkan data perbandingan antara ketiga perusahaan distribusi.

Kata kunci: data warehouse, konsolidasi, business intelligence

PENDAHULUAN

Data warehouse adalah kumpulan dari database yang terintegrasi dan berorientasi subjek serta dirancang untuk mendukung fungsi pengambilan keputusan, di mana setiap unit data relevan terhadap satu kejadian pada waktu tertentu (Inmon, 2005). Data warehouse dapat meningkatkan posisi daya saing perusahaan jika memiliki ketiga faktor berikut: mendefinisikan kebutuhan bisnis secara jelas, manajemen proyek, dan keuntungan bisnis yang terukur (Hwang dan Xu, 2007). Saat ini sudah banyak perusahaan yang telah memanfaatkan *data warehouse* dalam menyajikan laporan maupun menunjang keputusan bisnis. PT “A” yang telah memanfaatkan data warehouse untuk ketiga perusahaan distribusi, mendapatkan manfaat bagi kepentingan bisnis yang dijalankan. Pimpinan perusahaan PT “A”, membutuhkan perangkat yang dapat digunakan untuk melakukan analisis terhadap kondisi ketiga perusahaan distribusi, melakukan perbandingan dengan cepat dan mudah, sehingga dapat membuat kebijakan bisnis untuk keunggulan bersaing. Perangkat yang dapat digunakan langsung oleh pihak manajemen untuk mendapatkan “*insight*” dari data warehouse.

Guanm, Nunez dan Welsh (2002) menyatakan bahwa ketika pemakai berinteraksi dengan *data warehouse* memanfaatkan komponen presentasi, komponen presentasi adalah komponen yang terlihat ketika pemakai berinteraksi dengan *data warehouse*. Komponen ini biasa diimplementasikan dengan perangkat yang disebut perangkat *Business Intelligence*. Komponen presentasi digambarkan dengan tingkatan layanan yang berbeda dari laporan sederhana sampai dengan fungsi *roll-up* dan *drill-down* sampai analisis multidimensi sampai dengan kemampuan query *ad hoc*. Data Warehouse Institute (TDWI) mendefinisikan *business intelligence* sebagai proses, teknologi dan alat bantu yang dibutuhkan untuk mengubah data menjadi informasi, informasi menjadi pengetahuan, dan pengetahuan menjadi rencana yang dapat memicu aksi keuntungan dalam bisnis. *Business intelligence* terdiri dari *data warehouse*, alat bantu analisis bisnis, dan konten/manajemen pengetahuan (Loshin, 2003, p.6).

Manajemen PT “A” membutuhkan perangkat yang dapat membantu dalam melakukan analisis terhadap kondisi distribusi produk, di mana saat ini distribusi melalui tiga perusahaan distribusi yakni PT. “X”, “Y” dan “Z”. Masing-masing distributor memiliki data warehouse yang digunakan untuk menghasilkan laporan bagi manajemen PT “A”. Untuk mempercepat dalam pengambilan keputusan yang bersifat strategis maupun taktis, dibutuhkan integrasi (konsolidasi) ketiga data *warehouse* dan pembangunan aplikasi untuk kebutuhan analitis.

METODE

Penelitian ini menggunakan metode analisis untuk mengetahui kebutuhan informasi yang akan dihasilkan oleh *data warehouse*. Metode konsolidasi data warehouse dilakukan dengan membuat rancangan *snowflake*, *metadata*, integrasi data, *cube* dan antarmuka grafikal *business intelligence*.

HASIL DAN PEMBAHASAN

Analisis Kebutuhan Informasi

Untuk menunjang pengambilan keputusan yang bersifat strategis maupun taktis bagi manajemen PT “A”, informasi perlu disajikan dalam bentuk grafikal (dashboard) dalam aplikasi *business intelligence*.

Kebutuhan pengambilan keputusan terbagi ke dalam tiga area yakni penjualan, persediaan dan retur. Sebagai tahap awal dilakukan identifikasi kebutuhan informasi pada ketiga area tersebut, identifikasi didapatkan melalui proses bisnis yang berjalan dan informasi yang dibutuhkan manajemen. Berikut beberapa informasi atau laporan yang diidentifikasi: (1) laporan yang berhubungan dengan penjualan produk (obat) pada distributor “X”, “Y”, dan “Z”; yang terdiri dari total keseluruhan penjualan produk, jenis produk yang terjual, dan sepuluh produk yang paling banyak terjual; (2) laporan yang berhubungan dengan persediaan produk (obat) pada distributor “X”, “Y”, dan “Z”; terdiri dari total keseluruhan produk yang tersedia, dan jumlah produk yang tersedia; (3) laporan retur produk (obat) pada distributor “X”, “Y” dan “Z”; terdiri dari total produk yang diretur dan jumlah produk yang diretur.

Arsitektur Solusi Data Warehouse

Solusi arsitektur data warehouse dapat dilihat pada Gambar 1. Arsitektur Solusi *Data Warehouse* terdiri atas beberapa komponen yang saling berhubungan. Masing-masing dijelaskan sebagai berikut:

Sumber Data

Sumber data yang digunakan untuk proses konsolidasi adalah data warehouse pada distributor “X”, “Y”, dan “Z”. Saat ini data warehouse yang ada menggunakan SQL Server 2008, sehingga format file yang digunakan dari SQL Server 2008 dengan menggunakan format *.bak*.

ETL (Extract, Transform, and Load)

Proses ETL yang dilakukan adalah melakukan ekstraksi dari masing-masing *data warehouse* ke dalam *staging area* untuk kebutuhan integrasi data, setelah itu dilakukan proses transformasi data, transformasi dilakukan untuk melakukan penyamaan atau standarisasi di antaranya untuk tipe data, dan nama field, sedangkan load dilakukan untuk memindahkan data ke dalam *data warehouse*. ETLi data ini dilakukan dengan memakai sistem *SQL Server Integration Services* yang sudah disediakan oleh *Microsoft SQL Server 2008*.

Data Warehouse (dengan format multidimensional)

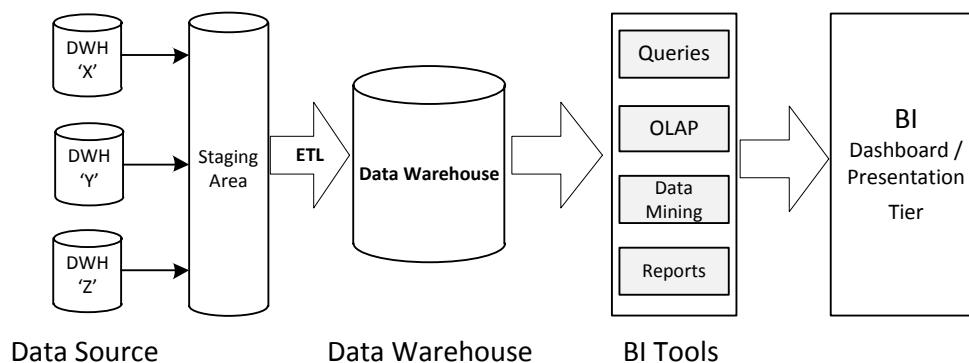
Data yang terdapat dalam data warehouse ini yang kemudian akan digunakan sebagai sumber laporan untuk kebutuhan pihak manajemen PT “A”.

OLAP (Online Analytical Processing)

OLAP yang digunakan adalah multidimensional OLAP. Setelah terbentuknya data warehouse, dan untuk mempermudah menampilkan data yang ada dalam berbagai dimensi, dalam bentuk “*Slide and dice*”. Dengan OLAP memudahkan bagi manajemen untuk menganalisis data dengan menggunakan *view* yang kompleks dan *multidimensional*. Dapat melihat data secara detail maupun secara agregat.

Dashboard

Untuk menyajikan informasi yang mudah dilihat dan dipahami digunakan antarmuka *Dashboard*, data ditampilkan dalam bentuk grafik, *chart*, dan diagram.



Gambar 1. Arsitektur solusi data warehouse.

Konsolidasi Data Warehouse

Konsolidasi data warehouse dirancang menggunakan pendekatan multidimensional, dan terbagi ke dalam tiga fakta, yaitu: penjualan, persediaan dan retur. Berikut fakta-fakta yang digunakan beserta dengan dimensi.

Snowflake Penjualan

Pada skema ini data tersebut dapat dilihat berdasarkan Dimensi Distributor, Dimensi DemandClass, Dimensi Warehouse, Dimensi SalesDiscType, Dimensi Customer, Dimensi CustomerGroup, Dimensi Channel, Dimensi Period, Dimensi SalesOrderType, Dimensi Product, Dimensi ProductGroup, Dimensi City, Dimensi Team, Dimensi ProductLine. Untuk *snowflake* penjualan (sales) dapat dilihat pada Gambar 2.

Snowflake Persediaan

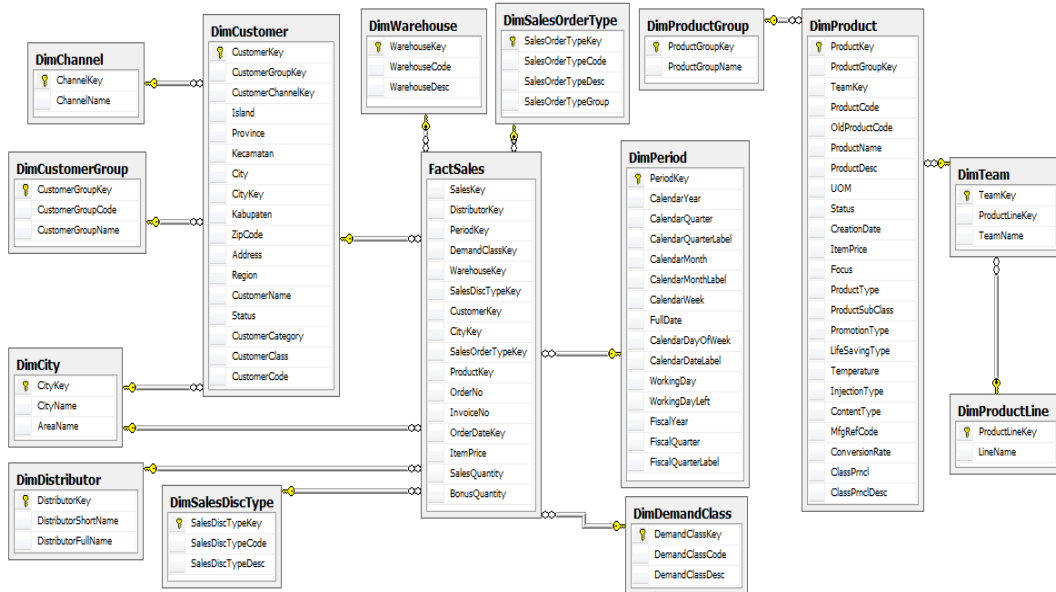
Pada skema ini data tersebut dapat dilihat berdasarkan Dimensi Distributor, Dimensi Period, Dimensi Product, Dimensi ProductGroup, Dimensi Warehouse, Dimensi City, Dimensi Team, Dimensi ProductLine.

Snowflake Retur

Pada skema ini data tersebut dapat dilihat berdasarkan Dimensi Distributor, Dimensi DemandClass, Dimensi Warehouse, Dimensi SalesDiscType, Dimensi Customer, Dimensi CustomerGroup, Dimensi Channel, Dimensi Period, Dimensi SalesOrderType, Dimensi Product, Dimensi ProductGroup, Dimensi City, Dimensi Team, Dimensi ProductLine.

Metadata

Metadata ini digunakan untuk melakukan pemetaan antara sumber data dengan target data (data warehouse "X", "Y" "Z" – data warehouse PT "A"). Pemetaan metadata disesuaikan dengan fakta yang digunakan yakni fakta penjualan, persediaan dan retur. Demikian pula dengan metadata untuk tabel dimensi.



Gambar 2. Snowflake fakta sales.

Integrasi Data

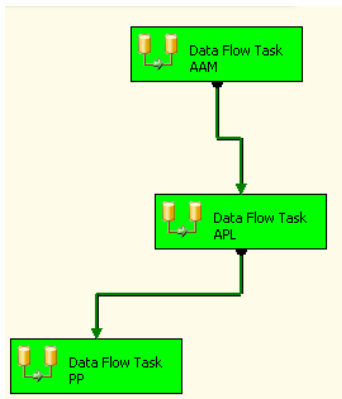
Dalam melakukan transformasi data warehouse pada ketiga distributor, digunakan *SQL Server 2008 Integration Services (SSIS)* yang terdiri dari komponen *control flow* dan *data flow*. *Control flow* adalah level tertinggi dari proses kontrol yang dapat digunakan untuk mengatur aktivitas proses *data flow* dan proses lainnya di dalam sebuah *package* yang didalamnya terdapat *task* dan *container*. *Task* adalah elemen *control flow* yang merupakan sebuah unit kerja. *Container* digunakan untuk mengelompokkan *task* menjadi unit kerja yang berguna dan dapat melakukan repetisi pada *control flow*.

Data flow task membungkus *data flow engine*, *engine* yang bekerja melakukan proses *ETL*. *Data flow engine* terdiri dari sumber (*source*), transformasi (*transformation*), dan tujuan (*destination*). *Source* adalah mendefinisikan lokasi sumber data. *OLE DB Source*: untuk mengambil data dari *OLE DB* seperti *SQL Server*, *Access*, *Oracle*, atau *DB2* sebagai contoh. *Transformation* adalah komponen kunci di dalam *data flow* yang mengubah data ke dalam format yang diinginkan atau digunakan untuk membersihkan dan melakukan standarisasi terhadap data. *Data Conversion* mengubah tipe data sebuah *column* menjadi tipe data yang lain. *Destination* menerima data dari *source* atau *transformation* untuk kemudian menyimpannya ke dalam data *source OLE DB* atau *file text*. *OLE DB Destination* menyimpan data ke koneksi *OLE DB* seperti *SQL Server*, *Oracle*, atau *Access*.

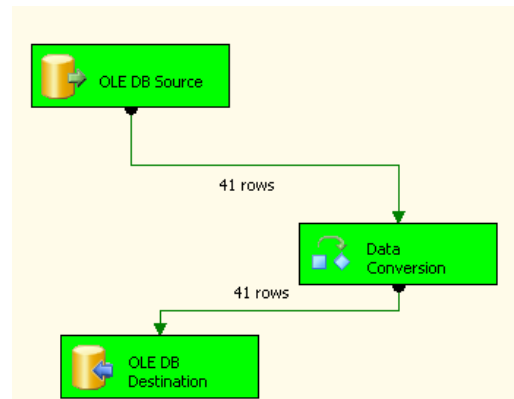
Proses transformasi data dimulai dari tabel *DimProductGroup*, *DimProductLine*, *DimTeam*, *DimCustomerGroup*, *DimWarehouse*, *DimSalesDiscType*, *DimSalesOrderType*, *DimDemandClass*, *DimCustomer*, *DimProduct*, *FactSales*, *FactInventory*, *FactReturn*. Beberapa contoh transformasi di antaranya: *DimProductGroup*. Pada *control flow* akan terlihat hasil eksekusi untuk setiap *task*, seperti terlihat pada Gambar 3.

Pada *data flow* “X” akan terlihat hasil eksekusi setiap komponen dan jumlah baris yang ditransformasi yaitu 41 rows, seperti terlihat pada Gambar 4. Pada *data flow* “Y” akan terlihat hasil eksekusi setiap komponen dan jumlah baris yang ditransformasi yaitu 51 rows, seperti terlihat pada

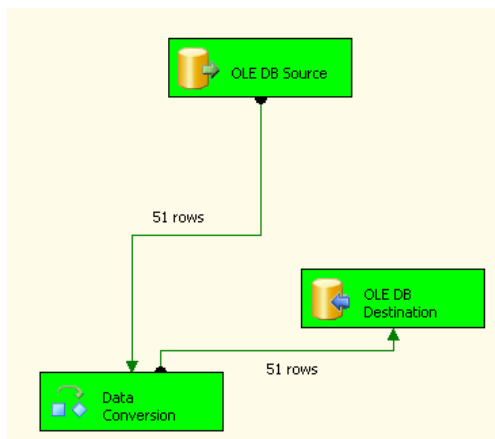
Gambar 5. Pada *data flow* “Z” akan terlihat hasil eksekusi setiap komponen dan jumlah baris yang ditransformasi yaitu 10 rows, seperti terlihat pada Gambar 6.



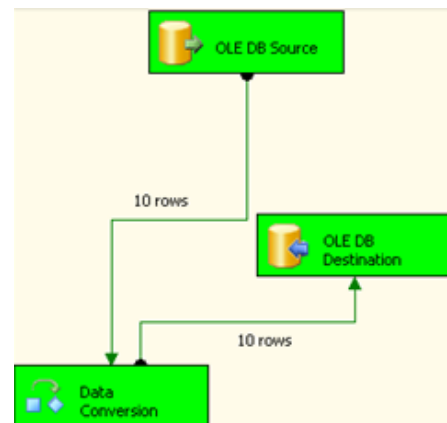
Gambar 3. Transformasi DimProductGroup.



Gambar 4. Transformasi Distributor “X”.



Gambar 5. Transformasi Distributor “Y”.



Gambar 6. Tranformasi Distributor “Z”.

OLAP CUBE

Dalam pembuatan OLAP CUBE tahapan yang dilakukan dapat dijelaskan sebagai berikut:

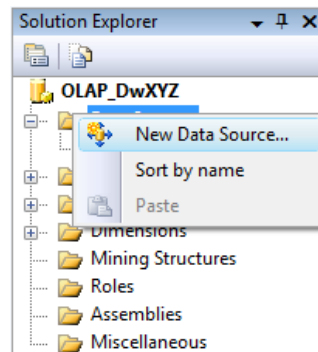
Membuat Data Source

Data *Source* menjadi acuan ke *database* mana project akan mengakses data. Provider yang digunakan adalah OLE DB (Gambar 7).

Membuat Koneksi

Dalam membuat koneksi untuk *data source*, ada tiga hal yang harus dilakukan, yaitu: (1) mendefinisikan koneksi – untuk membuat koneksi ke *data source*, perlu mendefinisikan *data source* yang dituju tersebut yaitu dengan mengklik tombol *new* pada *select how to define the connection*; (2) membuat koneksi dengan *Connection Manager* – pada layar *Connection Manager*, ketik (local) pada *Server Name*, lalu pilih DwXYZ pada combo *Select or enter a database name*; (3) memilih koneksi –

setelah muncul (local). USER-PC.DwXYZ pada layar *Select how to define the connection*, klik tombol *Next*.



Gambar 7. Membuat Data Source.

Membuat Data Source View

Data Source View (DSV) adalah *offline version* dari *metadata* yang merupakan gabungan dari tabel dan *view* yang di gunakan pada *Analysis Services project*.

Memberikan Relationship pada DimChannel

Untuk memberikan *Relation* pada *DimChannel* maka harus melakukan langkah – langkah seperti dibawah ini: Ekspan *DimChannel* → *Relationships* klik kanan → *New Relationships*

Memberikan Relationship DimChannel dengan DimCustomer

Setelah membuat *new Relationship* untuk *DimChannel*, selanjutnya mengatur *Specify Relationship* tersebut. Dengan pengaturan seperti berikut:

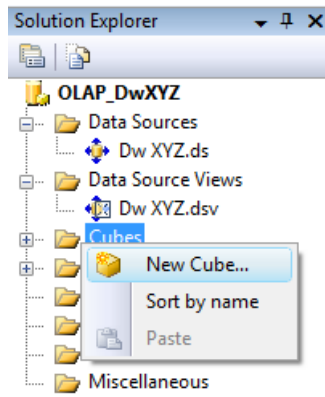
Source(foreign key) table: *DimCustomer*
Source Columns: *CustomerChannelKey*
Destination (primary key) table: *DimChannel*
Destination Columns: *ChannelKey*

Membuat Cube

Cube adalah adalah sebuah penyajian data secara *multidimensional* tergantung dari banyaknya nilai dalam suatu *Analysis* (Gambar 8).

Mengkonfigurasi Dimensi

Dimension designer mengizinkan untuk mengubah *properties*, menambahkan *attribute*, mendefenisikan *hierarchy* baru, dan mendefenisikan *attribute relationship*. Tab *Dimension Structure* dapat memilih *attribute* dari dimensi dan menyusun atribute tersebut dalam *hierarchy*. Tab *Attribute Relationship* dapat mengatur *relationship* antara *attribute*. *Relationship* ini membantu pada saat melakukan penelusuran member dari sebuah *hierarchy* dari dimensi tersebut.



Gambar 8. Membuat cube.

Proses konfigurasi dimensi dimulai dari konfigurasi DimDistributor, konfigurasi DimPeriod, Konfigurasi DimDemandClass, Konfigurasi DimWarehouse, Konfigurasi DimSalesDiscType, Konfigurasi DimCustomer, Konfigurasi DimCity, Konfigurasi SalesOrderType, Konfigurasi DimProduct. Salah satu contoh konfigurasi dimensi yang dijelaskan dalam ringkasan ini adalah:

Konfigurasi Dim Distributor

Pada DimDistributor dilakukan konfigurasi dan juga mengubah properties. Untuk mengubah *Properties* Dim Distributor:
 Pada *column Properties* → Name: Distributor Name, Name *column*: DistributorFullName.

Konfigurasi DimPeriod

Pada DimPeriod dilakukan konfigurasi, mengubah properties, membuat hierarchy, dan attribute relationships. Hierarchy yang dibuat disesuaikan dengan kebutuhan yang dibutuhkan oleh kebijakan perusahaan (Gambar 9).

Mengubah *Properties* Dim Period:

Pada *Attribute* Calendar Date

Pada *column Properties* → Name: Calendar Date, Name *column*: CalendarDateLabel.

Pada *Attribute* Calendar Year

Pada *column Properties* → Name: Calendar Year, Name *column*: CalendarYear.

Pada *Attribute* Fiscal Year

Pada *column Properties* → Name: Fiscal Year, Name *column*: FiscalYear.

Pada *Attribute* Calendar Month

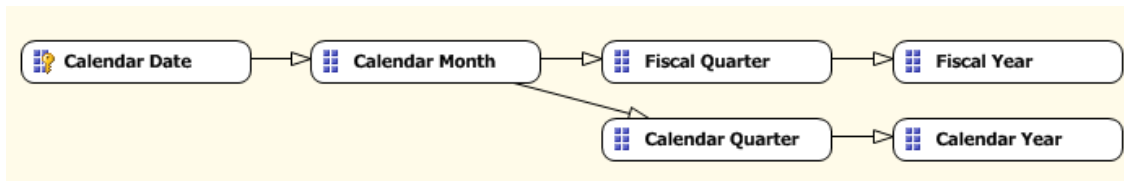
Pada *column Properties* → Name: Calendar Month, Name *column*: CalendarMonthLabel,
KeyColumns: CalendarMonth, FiscalYear, CalendarYear

Pada *Attribute* Calendar Quarter

Pada *column Properties* → Name: Calendar Quarter, Name *column*: CalendarQuarterLabel,
KeyColumns: CalendarQuarter, CalendarYear.

Pada *Attribute* Fiscal Quarter

Pada *column Properties* → Name: Fiscal Quarter, Name *column*: FiscalQuarterLabel,
KeyColumns: FiscalQuarter, FiscalYear.



Gambar 9. Perubahan *Attribute Relationships* DimPeriod.

Membuat *Hierarchy* DimPeriod: pada tab *Dimension Structure* di DimPeriod terdapat dua macam *hierarchy* yaitu *Calendar Hierarchy* dan *Fiscal Hierarchy*.

Attribute Relationships:

Untuk melakukan perubahan pada *Attribute Relationships*, maka pada *column attribute* harus mengklik kanan lalu pilih *New Attribute Relationships* pada:

Attribute Calendar Month: ubah *Related Attribute Name*: Calendar Quarter

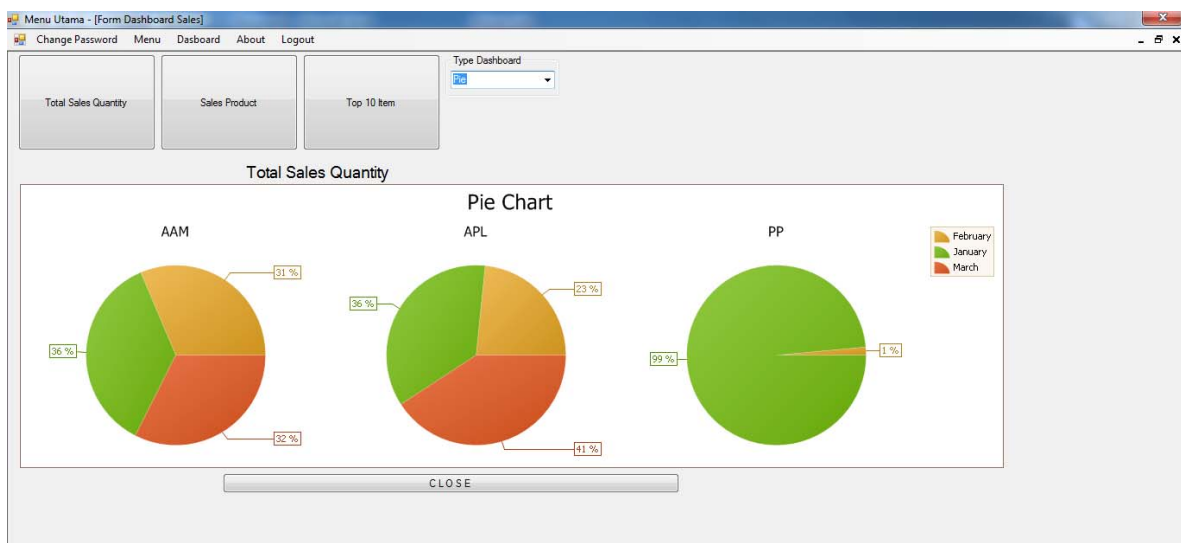
Attribute Calendar Month: ubah *Related Attribute Name*: Fiscal Quarter

Attribute Calendar Quarter: ubah *Related Attribute Name*: Calendar Year

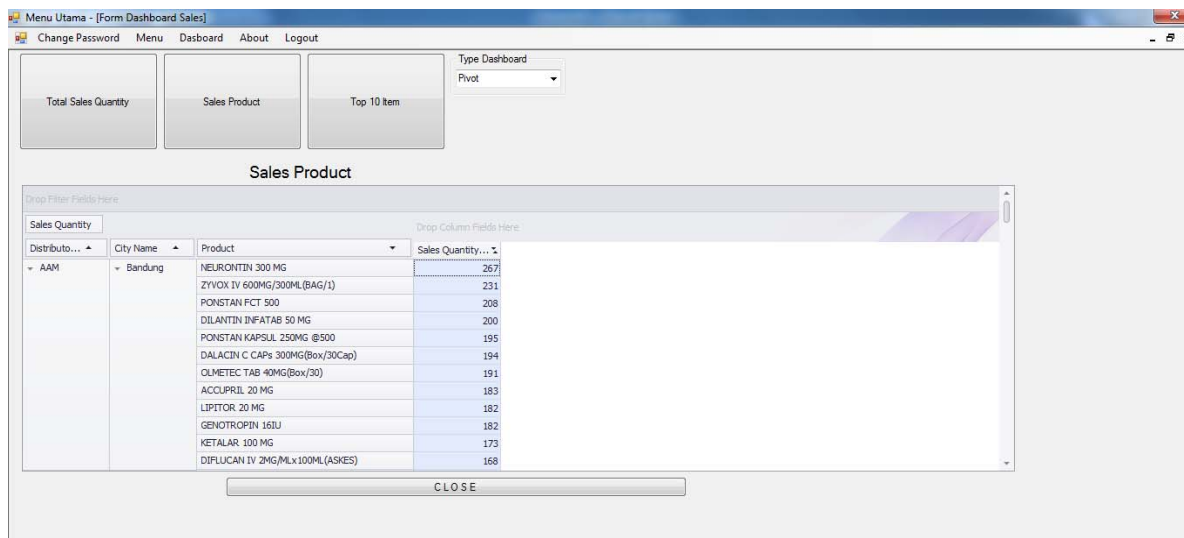
Attribute Fiscal Quarter: ubah *Related Attribute Name*: Fiscal Year.

Antarmuka Grafikal Business Intelligence

Tahapan akhir dalam proses ini adalah menampilkan data yang ada dalam bentuk grafis, di mana aplikasi business intelligence ini yang akan digunakan oleh pimpinan perusahaan untuk melakukan analisis terhadap kondisi ketiga perusahaan distribusi. Tampilan grafikal yang ada ini dibuat berdasarkan OLAP yang telah dibentuk diatas. Di bawah ini beberapa contoh antarmuka grafikal yang dapat disampaikan. Pada Gambar 10 terdapat antarmuka untuk menampilkan informasi total penjualan ketiga distributor menggunakan pie chart. Gambar 11 menyajikan informasi terkait dengan penjualan produk, ditampilkan dalam bentuk pivot table. Pada tampilan ini manajemen perusahaan dapat melihat informasi lebih detail, disinilah fungsi dari dimensi dapat dilihat dengan jelas, manajemen perusahaan dapat melihat data berdasarkan distributor, lokasi, dan jenis produk serta jumlah penjualan yang dilakukan. Dengan adanya antarmuka grafikal ini dan konsolidasi data warehouse yang dilakukan, pimpinan perusahaan dapat melihat seluruh informasi yang ada dalam sebuah antarmuka.



Gambar 10. Antarmuka grafikal total penjualan.



Gambar 11. Antarmuka grafikal penjualan dengan pivot.

PENUTUP

Proses konsolidasi data warehouse dilakukan sama seperti melakukan proses pembangunan data warehouse. Dengan adanya konsolidasi data warehouse, manajemen PT “A” dapat mengakses informasi untuk kebutuhan analisis dalam menentukan keputusan strategis dan taktis dalam sebuah aplikasi *business intelligence*. Aplikasi *business intelligence* ini bersumber dari tiga buah *data warehouse* yang diintegrasikan menjadi sebuah *data warehouse*.

Proses pembangunan *data warehouse* didahului dengan identifikasi kebutuhan informasi yang akan disajikan pada aplikasi *business intelligence*, kebutuhan informasi dijadikan dasar untuk pembuatan multidimensional skema dalam bentuk *snowflake*. Dari *snowflake* ini informasi yang akan ditampilkan dalam aplikasi *business intelligence* lebih jelas terlihat.

DAFTAR PUSTAKA

- Guan, J., Nunez, W., & Welsh, J.F. (2002). Institutional Strategy and Information Support: The Role of Data Warehousing in Higher Education. *Campus-Wide Information Systems*, 19 (5), ISSN 1065-074.
- Hwang, Mark I., & Xu, Hongjiang (2007). The Effect of Implementation Factors on Data Warehousing Success: An Exploratory Study. *Journal of Information Technology and Organizations*, 2. Diakses dari <http://jiito.org/articles/JIITOV2p001-014Hwang40.pdf>.
- Inmon, W.H. (2005). *Building the Data Warehouse* (4th edition). Indianapolis: Wiley.
- Loshin, David (2003). *Business Intelligence: the Savvy Manager's Guide*. San Francisco: Morgan Kaufmann.