

BAB II

LANDASAN TEORI

2.1 *Data Warehouse*

Data Warehouse (DW) merupakan basis data relasional yang didesain lebih kepada analisa dan *query* daripada proses transaksi, dan biasanya mengandung data historis dari proses transaksi dan bisa juga dari sumber lainnya untuk tujuan pengambilan keputusan. *Data Warehouse* dapat juga disebut sebagai tempat penyimpanan ringkasan dari data historis yang seringkali diambil dari basis data terpisah departemen, organisasi atau perusahaan (Kimball dan Caserta, 2004).

Menurut Inmon (2002) bahwa *data warehouse* merupakan koleksi data yang mempunyai sifat berorientasi subyek, terintegrasi, *time-variant*, dan bersifat tetap dari koleksi data dalam mendukung proses pengambilan keputusan manajemen.

Tujuan utama dari pembuatan *data warehouse* merupakan untuk menggabungkan data yang beragam ke dalam sebuah tempat penyimpanan dimana pengguna dapat dengan mudah menjalankan *query*, menghasilkan laporan, dan melakukan analisis. Meningkatkan efektifitas pembuatan keputusan merupakan salah satu keuntungan yang didapatkan dari *data warehouse*.

Dari definisi yang dijelaskan diatas, dapat disimpulkan bahwa *data warehouse* merupakan basisdata yang saling berinteraksi dan dapat digunakan untuk *query* dan analisis, bersifat orientasi subyek, terintegrasi, *time-variant*, tidak berubah (*adhoc*) yang nantinya digunakan dalam membantu pengambilan keputusan organisasi atau perusahaan oleh pihak pengambil keputusan.

2.1.1 Karakteristik *Data Warehouse*

Berikut ini merupakan karakteristik dari *data warehouse* (Inmon, 2002).

1. Berorientasi Subyek (*Subject Oriented*) yaitu *data warehouse* dirancang untuk menganalisa data berdasarkan subyek tertentu dalam perusahaan atau organisasi, bukan pada proses atau fungsi aplikasi tertentu. Hal ini disebabkan karena kebutuhan dari *data warehouse* adalah untuk menyimpan data yang digunakan sebagai penunjang suatu keputusan.

Tabel 2. 1 Perbandingan Fungsi Data Operasional dan *Data Warehouse*

Data Operasional	<i>Data Warehouse</i>
Dirancang hanya berorientasi pada aplikasi dan fungsi tertentu	Dirancang berdasar pada subyek-subyek tertentu (utama)
Fokusnya pada desain basisdata dan proses	Fokusnya pada pemodelan data dan desain data
Berisi rincian atau detail data	Berisi data histori yang akan dipakai dalam proses analisis
Relasi antar tabel berdasar aturan	Banyak aturan bisnis dapat tersaji

terkini (selalu mengikuti rule(aturan) terbaru)	antara tabel-tabel
---	--------------------

Sumber : (Inmon, 2002).

2. Terintegrasi (*Integrated*) yaitu *data warehouse* dapat menyimpan data yang berasal dari sumber data yang berbeda ke dalam suatu format yang konsisten dan saling terintegrasi satu dengan lainnya. Dengan demikian data tidak bisa dipecah-pecah karena data yang ada merupakan suatu kesatuan yang menunjang keseluruhan konsep *data warehouse* itu sendiri. Syarat integrasi sumber data dapat dipenuhi dengan berbagai cara seperti penamaan variabel yang konsisten, ukuran variabel yang konsisten, struktur pengkodean yang konsiten, dan atribut fisik dari data yang konsisten.
3. Rentang Waktu (*Time-Variant*) yaitu seluruh data pada *data warehouse* dapat dikatakan akurat atau valid pada rentang waktu tertentu. Untuk melihat interval waktu yang digunakan dalam mengukur keakuratan suatu *data warehouse* dapat menggunakan cara yaitu sebagai berikut.
 - a. Cara yang paling sederhana merupakan menyajikan *data warehouse* pada rentang waktu tertentu, misalnya antara 5 sampai 10 tahun ke belakang.
 - b. Cara yang kedua, dengan menggunakan variasi/perbedaan waktu yang disajikan dalam *data warehouse* baik secara implisit maupun secara eksplisit dengan unsur waktu dalam hari, minggu, bulan. Secara implisit misalnya pada saat data tersebut diduplikasi pada setiap akhir bulan, atau per tiga bulan. Unsur waktu akan tetap ada secara implisit di dalam data tersebut.

- c. Cara yang ketiga, variasi waktu yang disajikan *data warehouse* melalui serangkaian *snapshot* yang panjang. *Snapshot* merupakan tampilan dari sebagian data tertentu sesuai keinginan pemakai dari keseluruhan data yang ada bersifat *read-only*.
4. *Non Volatile* yaitu data yang ada pada *data warehouse* tidak dapat diperbaharui atau di *update*, tetapi hanya dapat di *refresh* dari data operasional atau sumber data berdasarkan waktu yang telah ditentukan. Data yang baru selalu ditambahkan sebagai suplemen bagi *database* itu sendiri dari pada sebagai sebuah perubahan. *Database* tersebut secara kontinyu menyerap data baru ini, kemudian secara *incremental* disatukan dengan data sebelumnya.

2.1.2 Tugas *Data Warehouse*

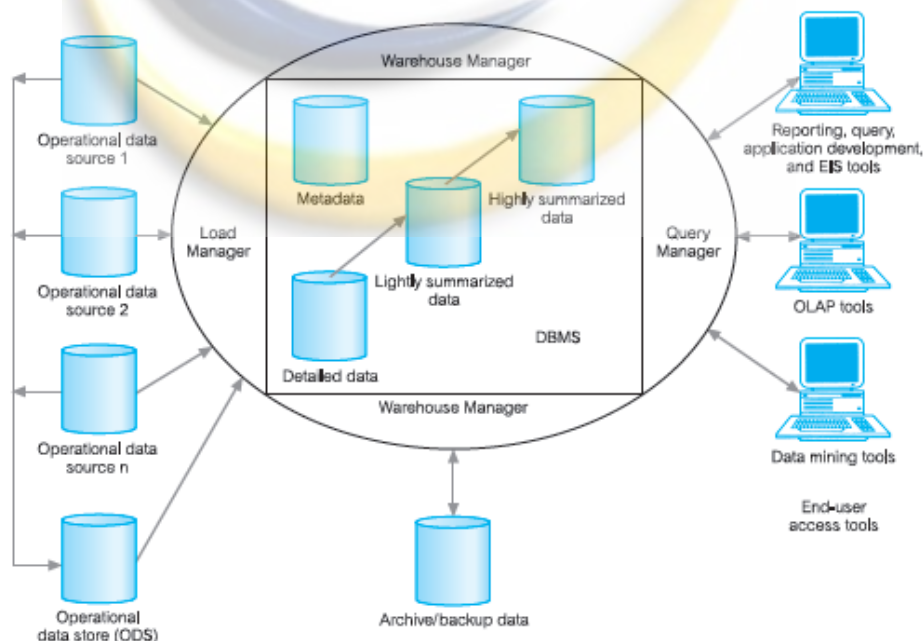
Ada empat tugas yang bisa dilakukan oleh *data warehouse* (Kimball dan Caserta, 2004) yaitu sebagai berikut.

- a. Pembuatan Laporan yakni proses pembuatan laporan merupakan salah satu kegunaan *data warehouse* yang paling umum dilakukan. Dengan menggunakan *query* sederhana didapatkan laporan perhari, perbulan, pertahun atau jangka waktu kapan pun yang diinginkan.
- b. OLAP yakni dengan adanya *data warehouse*, semua informasi baik detail maupun hasil *summary* yang dibutuhkan dalam proses analisa mudah di dapat. OLAP mendayagunakan konsep multidimensional dan memungkinkan

para pemakai menganalisa data sampai mendetail, tanpa mengetikkan satupun perintah *query*.

- c. *Data Mining* yakni merupakan proses untuk menggali pengetahuan dan informasi baru dari data yang berjumlah banyak pada *data warehouse*, dengan menggunakan kecerdasan buatan (*artificial intelligence*), statistik, dan matematika.
- d. Proses Informasi Eksekutif yakni *data warehouse* dapat membuat ringkasan informasi yang penting dengan tujuan membuat keputusan bisnis, tanpa harus menjelajahi keseluruhan data. Dengan menggunakan *data warehouse* segala laporan telah diringkas dan dapat pula mengetahui segala rinciannya secara lengkap, sehingga mempermudah proses pengambilan keputusan.

2.1.3 Arsitektur *Data Warehouse*



Gambar 2. 1 Arsitektur *Data Warehouse* (Connolly dan Begg, 2005:1157)

Komponen-komponen yang ada dalam arsitektur diatas antara lain sebagai berikut.

1. *Operational Data*

Operational Data adalah data yang digunakan dalam proses operasional harian.

2. *Operational Data Store (ODS)*

Operational Data Store (ODS) adalah sebuah tempat penyimpanan data saat ini dan data operasional terintegrasi yang digunakan untuk analisis. ODS menyediakan kemudahan untuk menggunakan *database* relasional ketika berada jauh dari fungsi *decision support* dari *data warehouse*. Dengan terlebih dahulu membangun ODS, pembuatan *data warehouse* menjadi lebih sederhana karena ODS dapat menyuplai data yang telah diekstrak dari sistem sumber.

3. *Load Manager*

Load manager disebut juga komponen *front-end* melakukan semua operasi yang berhubungan dengan ekstraksi dan *loading* data ke dalam *data warehouse*.

4. *Warehouse Manager*

Warehouse manager melaksanakan semua operasi yang berhubungan dengan pengelolaan data dalam *data warehouse*. Operasi-operasi yang dilaksanakan oleh *warehouse manager* meliputi: analisis data untuk memastikan konsistensinya; transformasi dan penggabungan sumber data dari tempat penyimpanan sementara ke dalam tabel-tabel *data warehouse*; pembuatan *index* dan *view* pada *base tables*; melakukan denormalisasi; melakukan *aggregation*; *backup* dan *archive data*.

5. *Query Manager*

Query manager menampilkan semua operasi yang terkait dengan *query* pengguna. Operasi yang ditampilkan oleh komponen ini meliputi mengarahkan *query* pada tabel yang cocok dan menjadwalkan pelaksanaan *query*.

6. *Detailed Data*

Dalam *data warehouse*, area ini menyimpan *detailed data* dalam skema *database*. Umumnya, *detailed data* tidak disimpan secara *online* tetapi dapat dibuat dengan mengagregasi data ke tingkatan yang lebih detail.

7. *Lightly and Highly Summarized Data*

Area ini menyimpan semua data *lightly* dan *highly summarized* yang sudah terdefinisi sebelumnya yang dibuat oleh *warehouse manager*. Tujuan informasi yang terangkum ini untuk meningkatkan performa *query*.

8. *Archive* atau *backup data*

Area ini menyimpan semua *detail* dan ringkasan data untuk tujuan *archiving* dan *backup*. Walaupun ringkasan data yang dihasilkan dari *data detail*, dengan itu akan memungkinkan *backup* ringkasan data yang dilakukan secara *online*. Jika data ini disimpan melebihi periode penyimpanan untuk data yang rinci. Data akan ditransfer ke *storage archives* seperti *optical disk*.

9. *Metadata*

Metadata merupakan data mengenai data yang mendeskripsikan *data warehouse*. *Metadata* digunakan untuk membangun, memelihara, mengatur, dan menggunakan *data warehouse*. *Metadaata* mengandung lokasi dan deskripsi dari komponen-komponen dalam *data warehouse*. Seperti nama, definisi, struktur, dan

isi dari *data warehouse* dan *end user view*. *Metadata* juga mengidentifikasi sumber-sumber data yang terintegrasi dan bertransformasi dalam *data warehouse*.

10. *End User Access Tools*

Pembangunan *data warehouse* adalah untuk menyediakan data yang konsisten kepada *user* yang akan digunakan untuk menganalisis dan menyediakan informasi untuk mendukung pengambilan keputusan.

2.1.4 Perancangan *Data Warehouse*

Salah satu dari metodologi perancangan *data warehouse* adalah *Nine-Step Methodology* oleh Connolly dan Begg (2005:1187) yang memiliki 9 (sembilan) langkah sebagai berikut.

1. Memilih Proses (*Choosing The Process*)

Proses (fungsi) mengacu pada subjek masalah dari *data mart* tertentu. *Data mart* yang pertama kali dibangun haruslah tepat waktu, sesuai dengan anggaran, dan dapat menjawab pertanyaan pertanyaan bisnis yang penting.

2. Memilih Sumber (*Choosing The Grain*)

Memilih *grain* berarti memutuskan secara pasti apa yang dinyatakan oleh *record* dari tabel fakta. Hanya dengan telah terpilihnya *grain* untuk tabel fakta maka kita dapat mengidentifikasi dimensi. Keputusan *grain* untuk tabel fakta juga menentukan *grain* untuk setiap dimensi pada tabel fakta.

3. Mengidentifikasi dan Penyesuaian Dimensi (*Identifying and Conforming The Dimensions*)

Dimensi menentukan konteks untuk memberikan pertanyaan mengenai fakta-fakta dalam tabel fakta. Kumpulan dimensi yang dibangun dengan baik membuat *data mart* dapat dimengerti dan mudah untuk digunakan. Dimensi diidentifikasi dalam detail yang cukup untuk mendeskripsikan data seperti *client* dan *properties* dari *grain* yang tepat. Jika ada dimensi yang muncul dalam dua *data mart*, maka kedua *data mart* tersebut harus memiliki dimensi yang sama, atau salah satu *data mart* adalah subset matematis dari *data mart* yang lain. Hanya dengan cara ini dua *data marts* berbagi satu atau lebih dimensi pada aplikasi yang sama. Ketika sebuah dimensi digunakan oleh lebih dari satu *data mart* dan dimensi tersebut tidak disinkronisasikan antar *data mart* maka seluruh *data warehouse* akan gagal, karena dua *data mart* tidak dapat digunakan pada saat yang bersamaan.

4. Memilih Fakta (Choosing The Fact)

Grain dari tabel fakta menentukan fakta mana yang dapat digunakan pada *data mart*. Semua fakta harus dinyatakan berdasarkan tingkatan yang tersirat oleh *grain*. Fakta tambahan dapat ditambahkan ke dalam tabel fakta pada setiap waktu dengan catatan fakta tersebut konsisten dengan *grain* dari tabel.

5. Menyimpan Perhitungan Awal dalam Tabel Fakta (*Storing Pre-Calculation in The Fact Table*)

Setelah fakta dipilih, setiap fakta harus dikaji ulang untuk menentukan apakah ada kemungkinan untuk melakukan *pre-calculations*. *Pre-calculations* terjadi ketika fakta terdiri dari *statement* untung dan rugi.

6. Melihat Kembali Table Dimensi (*Rounding Out The Dimension Tables*)

Pada langkah ini, kita kembali mengkaji tabel dimensi dan menambahkan sebanyak mungkin deskripsi teks ke dimensi. Teks deskripsi haruslah seintuitif mungkin dan dapat dimengerti oleh pengguna. Kegunaan dari *data mart* ditentukan oleh cakupan dan sifat atribut pada tabel dimensi.

7. Memilih Durasi *Database* (*Choosing The Duration of Database*)

Durasi mengukur berapa lama tabel fakta dapat disimpan. Pada banyak perusahaan, ada ketentuan untuk melihat pada periode waktu yang sama satu atau dua tahun sebelumnya. Tabel fakta yang sangat besar akan mengakibatkan setidaknya dua masalah yang signifikan pada *data warehouse*. Pertama, bertambahnya kesulitan untuk menjadikan data lama yang semakin bertambah sebagai sumber. Semakin lama suatu data, semakin banyak masalah dalam membaca dan menginterpretasikan *file* lama tersebut. Kedua, kebutuhan dimensi menggunakan versi yang lama, bukan versi yang baru.

8. Menelusuri Perubahan dari Dimensi secara Perlahan (*Tracking Slowly Changing Dimension*)

Pada tahap ini, *data warehouse* memperhatikan proses dimensi yang semakin tua seiring dengan berjalannya waktu. Untuk itu perlu dilakukan *update* agar *data warehouse* selalu konsisten. Berikut ini terdapat tiga tipe dasar perubahan dimensi secara perlahan.

a. Tipe 1 yaitu perubahan data secara langsung atau *update* tabel dimensi.

- b. Tipe 2 yaitu perubahan data membentuk *record* baru dengan *surrogate key* yang berbeda.
 - c. Tipe 3 yaitu perubahan data akan membentuk atribut atau kolom baru pada tabel dimensi.
9. Memutuskan Prioritas *Query* dan Tipe *Query* (*Deciding The Query Priorities and The Query Models*)

Pada tahap ini dipertimbangkan masalah perancangan fisik (*physical design*). Masalah utama pada perancangan fisik yang mempengaruhi persepsi pengguna akhir dari *data mart* adalah urutan penyusunan tabel fakta pada *disk* dan adanya *pre-stored summaries* dan agregasi.

2.1.5 Pemodelan Dimensional

Menurut Connolly dan Begg (2005: 1183), pemodelan dimensional adalah sebuah teknik perancangan logical yang bertujuan untuk mempresentasikan data ke dalam sebuah standar, bentuk intuitif yang dapat diakses dengan performa yang tinggi. Setiap model dimensional terdiri dari sebuah tabel dengan sebuah *primary key* komposit yang disebut dengan tabel fakta, dan sekumpulan tabel yang lebih kecil yang disebut dengan tabel dimensi. Setiap tabel dimensi memiliki sebuah *primary key* (nonkomposit) sederhana yang berkorespondensi tepat dengan satu *key* komposit pada tabel fakta. Dengan kata lain, *primary key* dari tabel fakta terbuat dari dua atau lebih *foreign key*. Karakteristik dengan struktur yang seperti bintang ini disebut dengan *star schema* atau *star join*.

1. *Star Schema*

Star schema adalah sebuah model data dimensional yang memiliki sebuah tabel fakta di pusatnya, dikelilingi oleh tabel-tabel dimensi yang berisi referensi data yang biasanya dapat didenormalisasi. Skema bintang mengeksploitasi karakteristik dari data faktual, seperti fakta dihasilkan dari *event-event* yang terjadi di masa lampau, dan tidak akan berubah terlepas dari bagaimana tabel fakta dianalisis. Fakta yang paling berguna dalam tabel fakta adalah perhitungan *numeric*, atau “fakta” yang terjadi pada setiap *record*. Sedangkan tabel dimensi mengandung informasi tekstual yang deskriptif. Atribut pada tabel dimensi digunakan sebagai batasan-batasan dalam *query data warehouse*. *Star schema* dapat digunakan untuk mempercepat kinerja *query* dengan mendenormalisasikan data referensi ke dalam sebuah tabel dimensi. Denormalisasi cocok dilakukan ketika ada beberapa entitas yang berhubungan dengan tabel dimensi sering diakses. Dengan denormalisasi maka penggabungan (*join*) tabel-tabel tambahan untuk mengakses atribut tersebut dapat dihindari.

2. *Snowflake Schema*

Snowflake schema adalah model data dimensional yang memiliki sebuah tabel fakta sebagai pusatnya, dikelilingi tabel-tabel dimensi yang ternormalisasi. *Snowflake schema* adalah sebuah variasi dari *star schema* dimana tabel dimensinya boleh memiliki dimensi. Penggunaan tabel dimensi pada *snowflake schema* sangatlah mendasar, sedangkan pada *star schema* tidak. *Snowflake schema* dibuat berdasarkan OLTP sehingga semua data akan termuat detail dalam setiap tabel fakta dan tabel dimensi.

3. *Starflake Schema*

Starflake schema adalah model data dimensional yang memiliki sebuah tabel fakta sebagai pusatnya, dikelilingi oleh tabel-tabel dimensi yang ternormalisasi dan terdenormalisasi. Beberapa pemodelan dimensional menggunakan campuran dari *star schema* yang terdenormalisasi dan *snowflake schema* yang ternormalisasi. Kombinasi dari *star schema* dan *snowflake schema* disebut *starflake schema*.

2.2 *Business Intelligence*

Business Intelligence atau disingkat BI merupakan kumpulan dan serangkaian kegiatan atau tahapan-tahapan untuk mengumpulkan data dan menganalisis data sehingga dapat digunakan untuk proses pengambilan keputusan yang lebih baik sehingga dapat digunakan dalam proses pengambilan keputusan vital dalam bisnis perusahaan atau keputusan untuk memperoleh tujuan dari bisnis perusahaan. *Business Intelligence* seringkali digunakan untuk dapat membantu kegiatan bisnis dalam proses pengambilan keputusan strategis seperti perencanaan anggaran tahunan, penentuan target *sales*, menganalisa dan memprediksi *trend* bisnis serta melakukan data konsolidasi untuk keperluan para pengambil keputusan.

Business Intelligence merupakan kerangka kerja konseptual untuk mendukung keputusan bisnis, *business intelligence* menggabungkan arsitektur, basis data atau *data warehouse*, *tool* analisis dan aplikasi (Turban dkk., 2007). *Business Intelligence* digunakan untuk aplikasi dan teknologi dalam

mengumpulkan, menyimpan, menganalisa, dan menyediakan akses pada data sehingga dapat membantu pengguna dari kalangan perusahaan atau organisasi untuk mengambil keputusan dengan lebih baik dan tepat (Brannon, 2010).

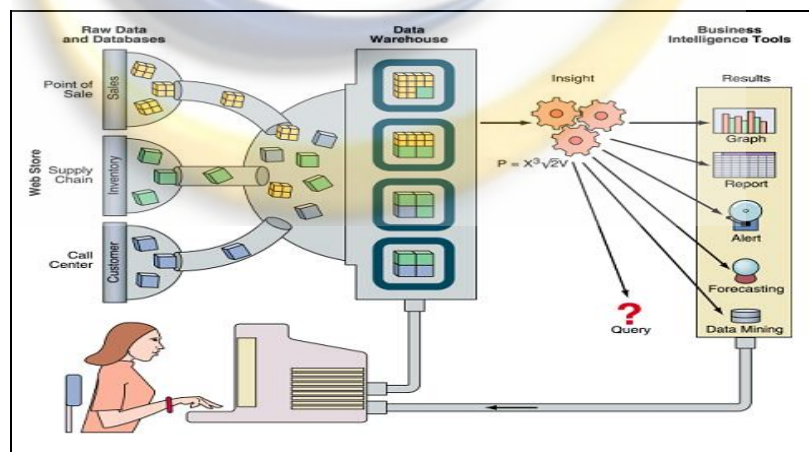
Business Intelligence menjelaskan tentang suatu konsep dan metode bagaimana cara atau prosedur untuk meningkatkan kualitas pengambilan keputusan bisnis berdasarkan sistem yang berbasis data-data dari berbagai data sumber. Dimana dalam proses *business intelligence* melakukan kegiatan pengambilan jumlah data yang besar, kemudian melakukan proses menganalisis data, dan dilanjutkan dengan menyajikan serta melaporkan hasil dari proses *business intelligence* tersebut sebagai bahan pertimbangan tindakan manajemen bisnis, memungkinkan untuk mengambil keputusan pokok bisnis saat dibutuhkan.

Selain itu, *business intelligence* digunakan pula untuk membuat referensi prediksi sehingga orang lain dapat mengukur dampak keputusan yang dibuat berdasarkan data yang ada. *Business Intelligence* juga merupakan proses yang berulang-ulang dengan cara menganalisis data untuk melihat apa yang terjadi. Kemudian mengambil tindakan untuk melanjutkan setiap hasil positif serta menghentikan jika ada potensi menuju arah negatif, sekaligus mengukur hasil dari setiap keputusan yang ada terhadap faktor pendukung eksternal maupun internal bagi bisnis perusahaan.

Dalam sebuah perusahaan atau organisasi, keputusan dibuat pada tingkatan atau level yang berbeda-beda, keputusan juga harus dilakukan secepat mungkin untuk mempertahankan daya saing, sehingga dapat membuat keputusan yang

benar dan menjadi dasar yang solid dalam hal data, informasi dan ketersediaan pengetahuan. Dasar ini dapat bersumber dari informasi atau data berita bisnis, konferensi, pelanggan, tenaga penjualan, dan sebagainya. Sehingga bersama-sama membentuk sebuah konsep mengenai pasar dan dunia bisnis tertentu.

Gambar 2.2 menyajikan pemahaman dasar mengenai sistem *business intelligence*. Sebuah sistem *business intelligence* dengan kata lain merupakan kombinasi *data warehouse* dan sistem pendukung keputusan. Hal ini menjelaskan bagaimana data dari sumber-sumber yang berbeda dapat diekstraksi dan disimpan dan selanjutnya diambil untuk dianalisis. Kegiatan utama *business intelligence* meliputi pengumpulan, menyiapkan dan menganalisa data. Dalam proses *business intelligence* data yang digunakan harus berkualitas tinggi, dengan cara memperolehnya dari berbagai sumber data yang dikumpulkan, kemudian diubah, lalu dibersihkan, selanjutnya dimuat dan disimpan dalam *data warehouse*.



Gambar 2. 2 Pemahaman Dasar Sistem *Business Intelligence* (Ranjan, 2009)

2.2.1 *Arsitektur Business Intelligence*

Arsitektur dari sebuah sistem *business intelligence* terdiri atas enam komponen utama (Vercellis, 2009) yaitu sebagai berikut.

1. *Data Source*

Pada tahap pertama ini diperlukan proses untuk mengumpulkan dan mengintegrasikan data yang disimpan dalam berbagai sumber yang bervariasi yang saling berbeda baik itu asal maupun jenisnya. Sumber data ini berasal dari data yang terdapat pada *operational system*, tetapi juga bisa berasal dari dokumen yang tidak terstruktur seperti email dan data yang dikirimkan oleh pihak luar.

2. *Data Warehouse*

Pada tahap ini proses menggunakan *extraction* dan *transformation tool* yang dikenal sebagai ETL (*Extract, Transform, Load*), data yang berasal dari berbagai sumber yang berbeda disimpan ke dalam basisdata yang ditujukan untuk mendukung proses analisis *business intelligence*.

3. *Data Exploration*

Pada tahap ini, *tools* yang berfungsi untuk keperluan analisis *business intelligence* pasif digunakan. *Tools* ini terdiri dari *query* dan *reporting system*, serta *statistical methods*. Metodologi ini bersifat pasif dikarenakan para pengambil keputusan harus mengambil keputusan berdasarkan hipotesis mereka sendiri atau mendefinisikan kriteria dari *data extraction*, kemudian menggunakan *tools* analisis untuk menemukan jawaban dan mencocokkannya dengan hipotesa awal mereka.

4. *Data Mining*

Pada tahap ini proses terdiri sejumlah metodologi *business intelligence* bersifat aktif yang tujuannya untuk mengekstrak informasi dan pengetahuan dari data tersebut. Metodologi ini berisi sejumlah model matematika untuk pengenalan pola (*pattern*), pembelajaran mesin (*machine learn*), dan teknik *data mining*.

5. *Optimization*

Pada tahap ini menghasilkan solusi dimana solusi terbaik harus dipilih dari sekian solusis alternatif yang ada, dan biasanya sangat banyak dan beragam atau bervariasi.

6. *Decisions*

Pada tahap ini yang menjadi persoalan utama merupakan bagaimana menentukan keputusan akhir yang akan diambil yang dikenal sebagai *decision making process*. Walaupun metodologi *business intelligence* berhasil diterapkan, pilihan untuk mengambil sebuah keputusan tetap ada ditangan para pengambil keputusan tersebut.

2.2.2 Jenis Business Intelligence

Menurut Turban, dkk (2007), *business intelligence* terbagi ke dalam lima jenis atau kategori yaitu sebagai berikut.

1. *Enterprise Reporting* yakni digunakan untuk menghasilkan laporan-laporan statis yang didistribusikan ke banyak orang. Jenis laporan ini sangat sesuai untuk laporan operasional dan *dashboard*.

2. *Cube Analysis* yakni digunakan untuk menyediakan analisis OLTP multidimensional yang ditujukan untuk manajer bisnis dalam lingkungan terbatas.
3. *Ad Hoc Query and Analysis* yakni digunakan untuk memberikan akses kepada *user* agar dapat melakukan *query* pada basis data, dan menggali informasi sampai pada tingkat paling dasar dari informasi transaksional. *Query* ini berfungsi untuk mengeksplor informasi yang dilakukan oleh *user*.
4. *Statistical Analysis and Data Mining* yakni digunakan untuk melakukan analisis prediksi atau menentukan korelasi sebab akibat diantara dua matrik.
5. *Delivery Report and Alert* yakni digunakan secara proaktif untuk mengirimkan laporan secara lengkap atau memberikan peringatan kepada populasi *user* yang besar atau banyak.

2.2.3 *Dashboard*

Dashboard adalah komponen yang umumnya memiliki *Performance Management Systems*, *Performance Measurement Systems*, *Business Process Management* (BPM) suites, dan *BI Platforms*. *Dashboard* menyediakan tampilan visual dari informasi penting yang disatukan dan diatur dalam sebuah layar tunggal sehingga informasi dapat dipahami cukup dengan sekali lihat, serta mudah untuk dieksplorasi.

Dashboard merupakan keturunan langsung dari *Executive Information Systems* (EIS) lama dan sistem *Decision Support Systems* (DSS), dengan meningkatkan fungsional dan penampilan. Karena mereka terhubung dengan sistem data yang kuat dan memanfaatkan *Key Performance Indicators* (KPI).

Menurut (Few, 2006:33) ada tiga jenis *dashboard*, yaitu sebagai berikut.

1. *Tactical Dashboard*

Mengukur produktivitas jangka pendek dan efektivitas. Hasilnya sering digunakan oleh kontributor individu.

2. *Operational Dashboard*

Mengukur efektivitas jangka pendek dari fungsi bisnis yang spesifik pada tim atau level unit bisnis. Level *dashboard* ini dapat secara potensial dikembangkan untuk seorang *knowledge worker* atau *local team manager*.

3. *Strategic Dashboard*

Dibangun untuk level pengaturan kebijakan dari organisasi. *Dashboard* menampilkan *metric* yang menggambarkan strategi dan tujuan korporasi.

2.3 Metode Visualisasi Data

Menurut Kirk (2012: 120), memilih metode visualisasi yang tepat merupakan hal yang sangat penting karena akan mempengaruhi definisi dari visual yang ditampilkan. Berikut adalah metode visualisasi data beserta fungsinya.

Tabel 2. 2 Metode Visualisasi Data

Metode Visualisasi	Fungsi
<i>Bar Chart</i>	Digunakan untuk membandingkan data yang memiliki kategori tertentu, dapat juga digunakan untuk menampilkan data yang kontinu dan diskontinu.
<i>Pie Chart</i>	Digunakan untuk memberikan rincian nilai kategori, merepresentasikan 100% dari kuantitas (jumlah dari nilai variabel pada seluruh kategori), dan ukuran dari

	setiap potongan adalah prosentase dari total yang direpresentasikan oleh kategori yang ditunjukkan potongan. <i>Piechart</i> biasanya digunakan untuk mempresentasikan frekuensi.
<i>Line Chart</i>	Digunakan untuk menampilkan informasi dalam rangkaian titik data yang dihubungkan dengan segmen garis lurus. <i>Line Chart</i> Digunakan untuk memvisualisasikan tren selama jangka waktu tertentu.
Tabel	Digunakan untuk mempresentasikan data dalam baris dan kolom dengan kategori tertentu, sehingga memudahkan dalam hal analisis. Tabel digunakan untuk menampilkan angka, tingkatan, proporsi, dan persentase kumulatif.

2.4 Borang Akreditasi

Borang adalah alat untuk mengumpulkan dan mengungkapkan data dan informasi yang digunakan untuk menilai kelayakan dan mutu institusi perguruan tinggi. Sedangkan akreditasi merupakan salah satu bentuk penilaian (evaluasi) mutu dan kelayakan institusi perguruan tinggi atau program studi yang dilakukan oleh organisasi atau badan mandiri di luar perguruan tinggi. Bentuk penilaian mutu eksternal yang lain adalah penilaian yang berkaitan dengan akuntabilitas, pemberian izin, pemberian lisensi oleh badan tertentu.