

BAB 2

LANDASAN TEORI

Pada Bab 2 ini akan diuraikan beberapa materi yang menjadi landasan teori dalam penelitian ini data *warehouse*, berbagai macam metode *data mining*, algoritma K-means, algoritma *apriori* serta materi berkaitan dengan produk *wireline* dan loyalitas pelanggan.

2.1 Data Warehouse

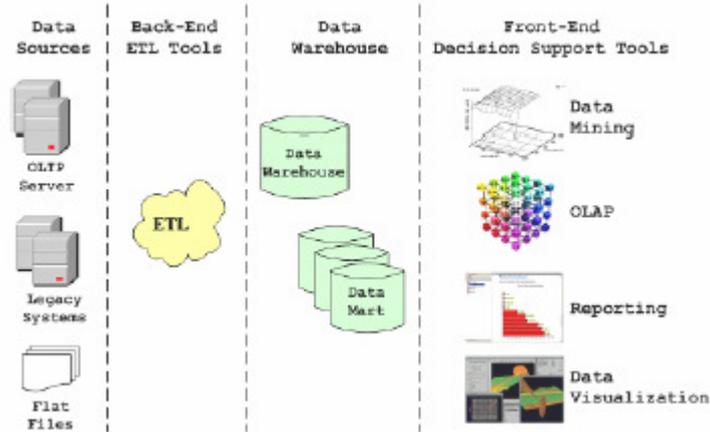
Perkembangan *data warehouse* saat ini telah dipahami peranannya sebagai sumber informasi dalam membuat keputusan dan berkaitan erat dengan *Decisions Support Systems (DSS)* (Wells, 2000). Data yang terdapat di *data warehouse* juga dapat dimanfaatkan dalam penyusunan perencanaan dan pengambilan keputusan strategis dengan menggunakan teknologi seperti; *On-Line Analytical Processing (OLAP)* dan *data mining* (Kontio,2006).

2.1.1 Arsitektur Data Warehouse

Agar didapat pemahaman yang cukup jelas mengenai arsitektur *data warehouse* langkah awal adalah dengan mengetahui apa definisi dari *data warehouse*. *Data warehouse* merupakan himpunan data yang berorientasi pada subjek, terintegrasi, *non-volatile* yang dapat mendukung manajemen untuk mengambil kebijakan (Inmon, 1993). Berikut ini dijelaskan berbagai macam arsitektur *data warehouse*.

A. Arsitektur *Data Warehouse* menurut Simitsis dan Theodoratos.

Bentuk dari arsitektur *data warehouse* menurut *Simitsis, Theooratus* ditunjukkan oleh gambar 2.1. Dalam gambar tersebut arsitektur *data warehouse* terbagi menjadi empat bagian, yaitu:



Gambar 2.1 Arsitektur *data warehouse* (Simitsis,Theodoratos,2006).

1. *Data Sources*

Data source diperoleh dari berbagai data transaksi dan produksi perusahaan hasil operasional aplikasi yang dijalankan tiap harinya. Data ini bisa tersimpan oleh *tools database* dan format struktur data yang mungkin berbeda, untuk itu diperlukan suatu bentuk pemodelan untuk pemahaman sesuai kebutuhan bisnis yang nantinya dilanjutkan ke proses berikutnya.

2. *Extraction, Transformation and Loading (ETL)*

Bagian ini terdiri dari tiga proses yang digabungkan menjadi satu. Proses *extraction* adalah proses yang dikerjakan untuk mendapatkan data dari sumber data. Proses *transformation* adalah proses format struktur data dari format sebelumnya. Proses *loading* adalah proses memasukkan data yang telah mengalami proses *extraction* dan *transformation* ke dalam *table* yang sudah dibuat.

Di proses ETL juga dilakukan pengerjaan seperti: pembersihan data, pengecekan duplikasi data, standarisasi data, dan penamaan konvensi data sesuai kebutuhan *data warehouse*. Dimana tingkat kualitas *data warehouse* yang dihasilkan sangat ditentukan diproses ini (Winkler,2006). Proses pengerjaan ETL biasanya dibantu oleh *tools* yang merupakan bagian dari arsitektur *data warehouse* (Simitsis, Theodoratos, 2006).

3. *Data Storage/Data Warehouse.*

Data warehouse sudah menjadi bagian terakhir pembentukan data, dimana data yang berada di *data warehouse* siap untuk digunakan untuk pemenuhan kebutuhan dalam mendukung pengambilan kebijakan dan keputusan strategis. *Data warehouse* bisa terdiri dari kumpulan beberapa *data mart* yang bersifat *dependence mart* dan *independence mart*.

4. *Front-End Decision Support Tools.*

Front-End Decision Support Tools adalah bagian pemanfaatan dari *data warehouse* yang telah dibuat, diantara lainnya menggunakan teknologi OLAP dan *data mining*.

B. *Arsitektur Data warehouse* menurut *Hoffer, Prescott* dan *McFadden*

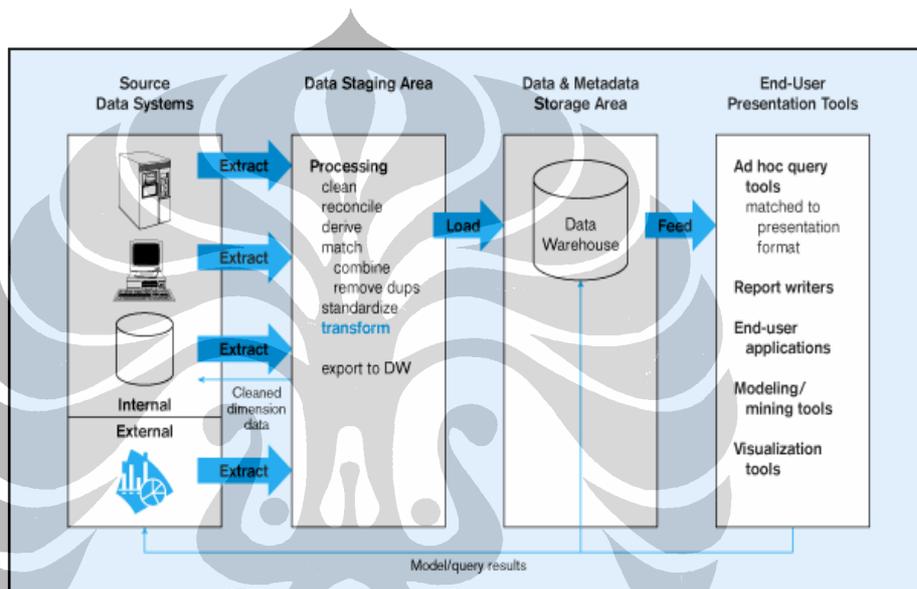
Bentuk arsitektur *warehouse* ini terdiri dari *Data source systems*, *Data staging area*, *data and metadata storage area* dan *End-user analysis tools*. Adapun secara visual bentuk arsitekturnya seperti yang ditunjukkan dalam gambar 2.2 . Kemudian penjelasan masing-masing elemen dari *data warehouse* adalah sebagai berikut :

- *Data source systems*
Data source systems ini terdiri dari data internal dan eksternal. Data internal merupakan data transaksional yang tersimpan secara harian. Sedangkan data eksternal merupakan yang berasal dari luar organisasi yang dibutuhkan untuk keperluan analisa berupa data dari vendor, internet dan data terkait lainnya.
- *Data staging area*
Merupakan penyimpanan sementara untuk data dari *sources system* yang akan dilakukan ekstraksi dan transformasi sesuai dengan format yang sesuai dan tepat untuk *data warehouse* ataupun *data marts*. Selain itu di dalam *data staging area* ini dilakukan proses pembersihan, rekonsiliasi, standarisasi dan *matching*.
- *Data and metadata storage area*
Merupakan penyimpanan secara fisik dari data yang sudah dilakukan transformasi maupun pengumpulan data untuk *data warehouse*. Bentuk data

biasanya sudah dalam bentuk *metadata* atau *multidimensional* data, yang nantinya sudah siap dilakukan untuk proses analisis oleh *end user*.

- *End-user analysis tools*

Merupakan berbagai macam *tools* yang digunakan oleh *end user* untuk melakukan *query*, menganalisa, membuat laporan ataupun mengeksplorasi informasi dan data yang tersimpan di dalam *data warehouse*. Biasanya dalam bentuk *ad hoc query tools*, *Report writer*, *Modelling/mining tools* dan *visualization tools*.



Gambar 2.2 Arsitektur *data warehouse* (Hoffer, Prescott, McFadden, 2005).

2.1.2 *Extraction, Transformation dan Loading (ETL)*

Merupakan proses pembuatan informasional data penyimpanan yang berasal dari data operasional. Sering kali disebut sebagai proses *Extract, Transform* dan *Load process*, atau dikenal dengan singkatan ETL. Terdapat empat tahapan dasar dalam proses ETL. (Hoffer, Prescott, McFadden, 2005).

Adapun tahapan tersebut adalah :

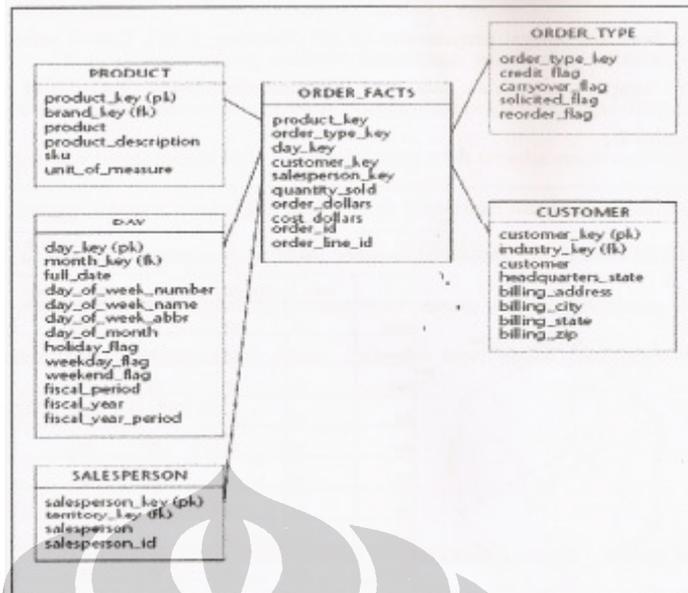
- *Capture/Extract source data*

Adalah proses mendapatkan potret data yang terpilih dalam subset sumber data untuk proses loading ke *datawarehouse*.

- *Cleanse (scrub)*
Adapun yang termasuk dalam proses ini adalah pembetulan *error* yang terjadi seperti *misspellings, erroneous dates, incorrect field usage, mismatched addresses, missing data, duplicate data, inconsistencies*. Termasuk juga proses *decoding, reformatting, time stamping, conversion, key generation, merging, error detection/logging, locating missing data*.
- *Transform*
Adalah proses konversi format data dari sistem data operasional menjadi data *warehouse*. Bentuk *transformation* ada dua macam yaitu *record level transformation (Selection-data partitioning, Joining – data combining, Aggregation – data summarization)* dan *Field-level transformation (single-field – from one field to one field, multi-field – from many fields to one, or one field to many)*.
- *Load and Index*
Adalah proses menempatkan data yang telah bertransformasi ke dalam *data warehouse* dan membuat indeksnya.. Terdapat dua mode untuk proses ini yaitu *Refresh mode: melakukan bulk rewriting terhadap target data secara periodik*) dan *Update mode: (hanya data yang berubah yang perlu dilakukan updating ke data warehouse)*.

2.1.3 Skema *Data Warehouse*

Tabel-tabel didalam *data warehouse* dirancang menggunakan pemodelan dimensi atau yang biasa disebut *star schema* dapat meningkatkan kinerja *data warehouse* yang dibuat. Komponen *star schema* di *data warehouse* dideskripsikan dengan menggunakan teknik pemodelan *table* fakta dan *table* dimensi. Keterkaitan pemodelan antara tabel fakta dan tabel dimensi hampir mirip dengan penggunaan konsep dari *Entity-Relationship* (ER) dengan memberikan beberapa pembatasan. Setiap model *star schema* terdiri dari satu tabel fakta dengan komposisi *primary key*, dan beberapa tabel dimensi dengan komposisi *foreign key* yang berelasi dengan *primary key* yang ada di tabel fakta (Connolly, 2005). Contoh sederhana dari *star schema* dapat dilihat pada gambar 2.4 dibawah ini.

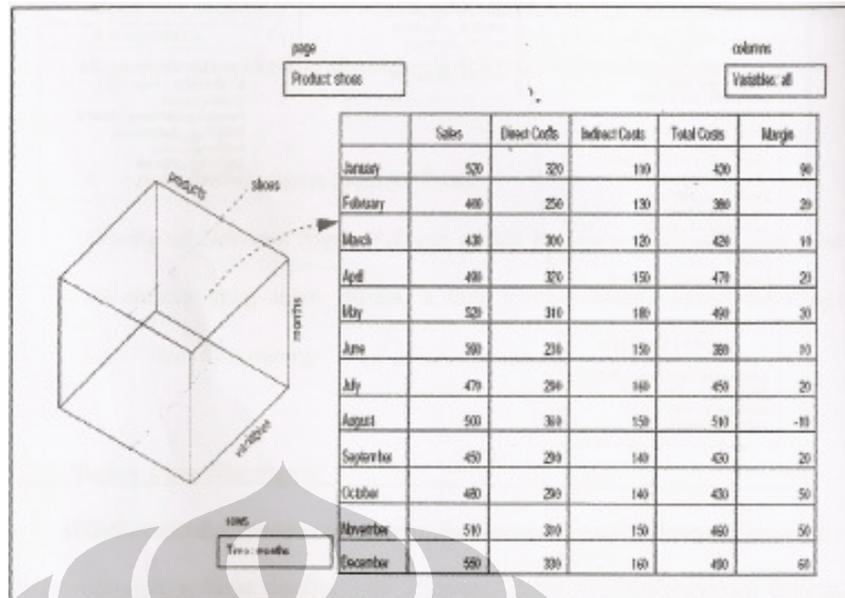


Gambar 2.3 *Star schema* untuk proses pemesanan (Adamson, 2006)

2.2 *On-Line Analytical Processing (OLAP)*

Kemampuan *On-Line Analytical Processing (OLAP)* dalam mendukung queries dan analisa data pada aggregation *database* yang ada di *data warehouse*, menjadikan OLAP sebagai system dengan kemampuan mengumpulkan, mengelola, memroses dan menghadirkan multidimensional data. Penerapan teknologi OLAP dapat memberikan informasi untuk penyusunan perencanaan dan pembuatan kebijakan bisnis perusahaan (Kontio, 2006), (Peter, 2006).

Ada dua metode utama OLAP yang banyak diterapkan yaitu: Relational OLAP (ROLAP) dan Multidimensional OLAP (MOLAP) (Boon-NoiTan, 2006). Kedua metode ini memanfaatkan pembentukan data dari *cube* atau multidimensional (Boon-NoiTan, 2006), dimana karakter data multidimensional ini dipercaya sebagai penyelesaian OLAP (Thomsen, 2002). Contoh sederhana dari tampilan pemodelan data *cube* dan multidimensional dapat dilihat pada gambar 2.5



Gambar 2.4 Bentuk pemodelan data *cube* dan tampilan data sisi dari *cube*
(Thomsen, 2002)

Bagian kiri gambar menunjukkan pemodelan data yang terbentuk dalam sebuah *cube*. *Cube* ini terdiri dari data *multidimensional product, months (time),* dan variabel. Di bagian tengah *cube* dibuat suatu irisan (sisi bidang) untuk *product shoes* yang kemudian ditampilkan menjadi halaman data yang terlihat di bagian kanan gambar.

2.3 Konsep Data Mining

Secara umum definisi *Data Mining* (penambangan data) dapat dijelaskan sebagai teknologi baru dalam pencarian informasi yang sangat bermanfaat bagi kepentingan bisnis perusahaan. Informasi yang diperoleh dari proses pencarian yang dilakukan dalam *data mining* harus memiliki arti yang sangat penting dan bernilai tinggi. Pendekatan yang dilakukan dalam *data mining* menggunakan *discovery based* dengan bantuan algoritma-algoritma untuk menentukan keterkaitan dari data yang dieksplorasi sehingga didapatkan pola-pola informasi. Di bawah ini beberapa definisi sebagai konsep *data mining* dari beberapa literatur yang terkait :

1. *Data Mining* adalah suatu proses pencarian pengetahuan. *Data mining* dapat membantu kita memahami substansi dari data secara khusus yang tadinya

tidak diketahui seperti pola-pola atau trend yang ada dalam data yang sebelumnya tidak pernah diketahui (Connolly, 2005).

2. *Data Mining* juga bisa dikatakan proses ekstraksi data yang valid, dimana sebelumnya ada informasi yang sangat berharga, lebih menyeluruh namun belum diketahui, hal ini diperlukan usaha yang lebih besar untuk menemukannya dari suatu *database* di mana informasi yang diperoleh dapat digunakan untuk membuat kebijakan bisnis yang penting (Ponniah, 2001)
3. *Data mining* merupakan suatu proses penyelidikan yang berlangsung secara besar-besaran dari suatu data untuk menemukan pola-pola yang tersembunyi dan menemukan pengetahuan baru (*Oracle Corporation*, 2006)
4. *Data Mining* adalah suatu proses penggunaan berbagai variasi dan *tools* analisa data untuk menemukan pola-pola dan keterkaitan dalam suatu data dimana hasilnya dapat digunakan untuk membuat prediksi yang valid (*Two Crows Corporation*, 2005)
5. *Data Mining* adalah suatu proses pencarian berarti mengenai hubungan, pola, dan trend yang baru dengan melakukan penyelidikan secara terus-menerus dari data yang berjumlah sangat besar di suatu media penyimpanan data. Dalam pencarian dan penggunaan pola-pola dengan memadukan teknologi yang sudah ada seperti teknologi statistik (Gatner Group, 2007)

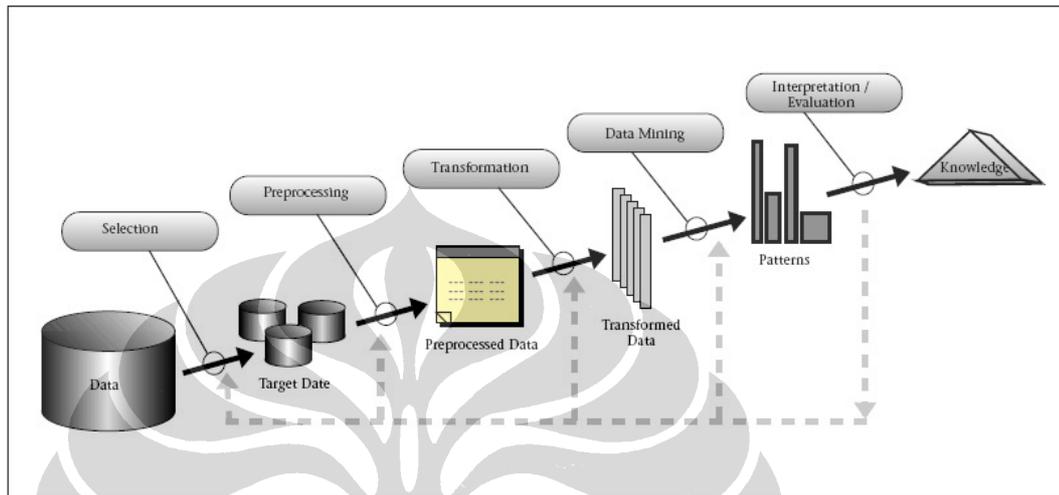
2.4 Metodologi *Knowledge Discovery in Database* (KDD)

Data mining bukanlah suatu teknologi yang berdiri sendiri melainkan suatu tahapan dari proses pencarian pengetahuan dalam *database* yang biasa dikenal dengan sebutan *Knowledge Discovery in Database* (KDD). Proses berulang *data mining* yang dilakukan dari tahapan-tahapan KDD bisa disebut juga suatu metodologi *data mining*.

Tahapan yang ada diproses KDD sampai saat ini banyak dikeluarkan oleh peneliti KDD. Berdasarkan kajian beberapa proses yang dilakukan, dapat diketahui bahwasannya tahapan yang ada antara satu proses KDD dengan lainnya tidak terjadi pertentangan yang prinsip. Perbedaan yang ditemukan adalah jumlah tahapan diproses KDD. Beberapa definisi dan tahapan dari proses KDD akan dibahas berikut ini.

2.4.1 Proses KDD Menurut Fayyad

KDD merupakan suatu proses interaktif dan iterative, meliputi banyak tahapan dengan berbagai keputusan yang dibuat oleh pemakai (Fayyad et.al, 1996). Tahapan-tahapan yang dilakukan dapat dilihat pada gambar 2.5



Gambar 2.5 Tahapan-tahapan dalam proses KDD (Fayyad et.al, 1996)

1. Pemahaman mengenai domain aplikasi dan relevansi terhadap pengetahuan yang ada serta mengidentifikasi tujuan dan sasaran yang ingin dicapai.
2. Membuat *database* sebagai data target: melakukan pemilihan data untuk membuat *database* atau memfokuskan penyediaan *database* untuk proses *discovery* yang dilakukan dengan menggunakan subset variabel atau sample data.
3. Pembersihan data : melakukan pembersihan data seperti menghilangkan noise, menjaga data dari ketidak konsisten atau tidak relevan serta melengkapi relasi *field* yang ada di *database*.
4. Memproyeksikan pengurangan data: disini sudah dimulia dilakukan pencarian fitur-fitur yang bermanfaat untuk dipersentasikan kedalam tujuan dan sasaran yang telah ditentukan.
5. Pemilihan teknik dari metode *data mining* untuk melakukan pencocokkan dari tujuan yang sudah ditentukan di tahap satu seperti: *segmentasi*, *classification*, *regresi* dan sebagainya.

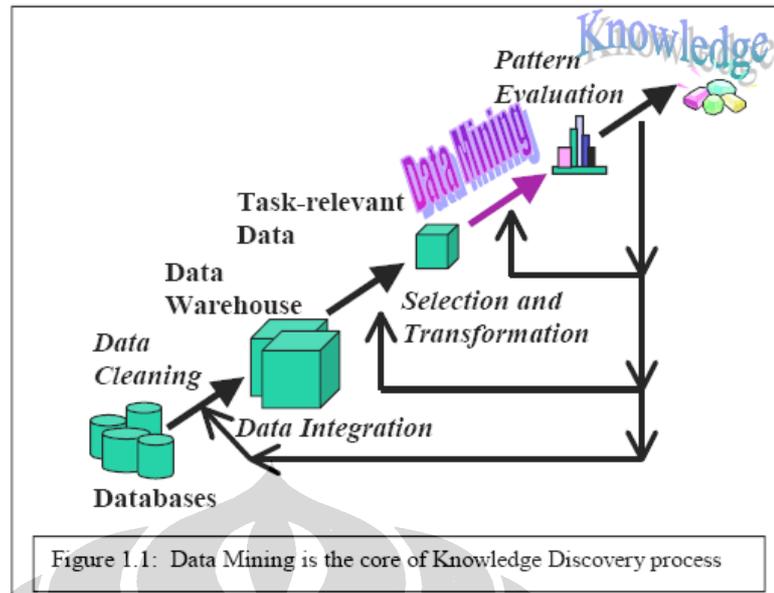
6. Melakukan eksplorasi dan pemodelan serta pemilihan hipotesis: pemilihan algoritma *data mining* dan metoda atau teknik untuk digunakan dalam pencarian pola-pola data.
7. *Data mining*: proses pencarian pola-pola data berharga yang diinginkan dari kumpulan data yang direpresikan.
8. Mengevaluasi dan penterjemahan pola-pola yang diperoleh dari *data mining* di tahapan ini ada kemungkinan dilakukan pengulangan daritahap satu sampai tujuh.
9. Menindaklanjuti hasil pencarian pola-pola sebagai pengetahuan baru dengan melakukan pengujian pada data berbeda, juga dilakukan penyelesaian apabila terjadi konflik antara pengetahuan yang diperoleh dengan pengetahuan yang dipercaya sebelumnya.

2.4.2 Proses KDD Menurut OSMAR

Osmar (1999), mendefinisikan *data mining* adalah bagian dari *knowledge discovery process* yang bertujuan mengekstrasi data implicit yang sebelumnya tidak diketahui informasinya, yang nantinya dapat dimanfaatkan untuk pengambilan keputusan. Adapun tahapan iterasi proses *discovery knowledge* sebagai berikut :

- *Data cleaning*
- *Data integration*
- *Data selection*
- *Data transformation*
- *Data mining*
- *Pattern evaluation*
- *Knowledge representation*

Adapun secara proses ditunjukkan dalam gambar 2.6 berikut ini:



Gambar 2.6 Proses KDD (OSMAR R. Zaïane, 1999)

2.4.3 Proses KDD Menurut Baritchi.

Baritchi (2004), mendefinisikan KDD adalah suatu proses ketekunan dalam pencarian pengetahuan di *database* yang benar tujuannya menggali pola-pola tersembunyi atau pengetahuan di dalam data. Berikut tahapan-tahapan yang ditetapkan:

- a. *Domain Analysis.*
- b. *Selection data.*
- c. *Preprocessing data.*
- d. *Transformation data.*
- e. *Data Mining*
- f. *Interpretation and evaluation.*
- g. *Consolidation of Discovered Knowledge.*

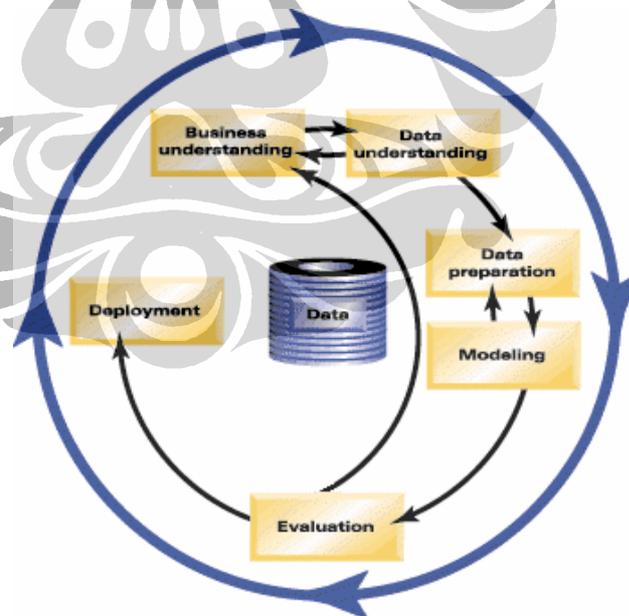
2.4.4 Proses KDD Cross-Industry Standard Process for Data Mining

Metodologi ini dibuat oleh beberapa orang dari sebuah konsorium perusahaan yaitu: Pete Champman (NCR) Julian Clinton (SPSS), Randy Kerber(NCR), Thomas Khabaza (SPSS), Thomas Reinartz (Daimler Chrysler), Colin Shearer (SPSS) dan Rudiger Wirth (Daimler Chrysler). Proses tahapan

KDD dalam metodologi ini dinamakan *Cross Of Industry Standard Process for Data Mining (CRISP-DM V.1.0, 2002)*. Di bawah ini tahapan-tahapan dari *CRISP-DM*:

1. *Business understanding phase.*
2. *Data understanding phase.*
3. *Data Preparation phase.*
4. *Modeling phase.*
5. *Evaluation phase.*
6. *Deployment phase.*

Metodologi CRISP-DM saat ini banyak digunakan sebagai referensi dalam *KDD*, ini dapat dilihat dari beberapa referensi yang menggunakan metodologi CRISP-DM (Larose, 2005, p.5-9), (Ye, 2003, p.490), (Connolly, Begg, 2005, p.1239-1240), (Tang,MacLennan, p28-30). Dalam penelitian yang menggunakan metodologi CRISP-DM. Dibawah ini proses iterasi dari tahapan KDD CRISP-DM, yang ditunjukkan oleh gambar 2.7



Gambar 2.7 Proses tahapan *KDD* CRISP-DM (*CRISP-DM V.1.0,2002*)

2.4.5 Proses KDD *Two Crows Corporation*

Metodologi KDD dari *Two Crows Corporation* dibuat dengan mengembangkan metodologi CRISP-DM. Terdapat satu tahapan di metodologi KDD *Two Crows Corporation* yang tidak ada di metodologi CRISP-DM, yaitu tahapan penyelidikan data sebelum dilakukan proses *data mining* dengan menggunakan teknologi OLAP. Karena adanya kombinasi teknologi OLAP diproses tahapan KDD. Penggunaan teknologi OLAP di tahapan proses KDD juga digunakan oleh peneliti KDD antara lain; oleh Silver (Silver 2005) dan Payton (Payton 2005). Adapun tahapannya adalah sebagai berikut :

1. *Define business problem*
2. *Build data mining database*
3. *Explore data*
4. *Prepare data for modeling*
5. *Build model*
6. *Evaluate model*
7. *Deploy model and results*

2.5 Teknik-Teknik *Data Mining*

Saat ini kategori teknik-teknik *data mining* dibagi dalam dua kegiatan utama yaitu: deskripsi dan prediksi (Baricthi, 2004), (*Two Crows Corporation*, 2005)

- Deskripsi

Deskripsi merupakan kegiatan yang paling banyak dilakukan di *data mining* untuk memperoleh pola data. Proses ini berfokus pada pencarian pola-pola berharga yang tersembunyi dalam *database* yang dieksplorasi. Teknik-teknik *data mining* yang termasuk dalam kategori ini diantara lain: *association*, *segmentasi*, *summaries and visualization*, dan sebagainya.

- Prediksi

Sasaran dari tugas dari Prediksi adalah melakukan prediksi dengan menggunakan nilai-nilai *variabel* atau *field-field* dimasa akan datang yang dibutuhkan, dimana informasi belum diketahui saat ini. Dalam konteks *KDD*, prediksi dianggap kurang penting dibandingkan dengan deskripsi. Teknik-teknik

data mining yang termasuk dalam kata kategori prediksi adalah: *classification, regression, time series* dan lain-lain.

2.5.1 Association Rules

Association rules adalah teknik *data mining* untuk menemukan aturan *associative* antara suatu kombinasi item yang ada didalam *database* antara satu item dengan item lainnya. *Association rules* terdiri dari dua komponen yang biasa disebut dengan *antecedent* dan *consequent* (Webb, 2003). *Association* dapat ditulis seperti $A \Rightarrow B$, dimana A disebut *antecedent or left hand side (LHS)*, dan B disebut *consequent or right hand side (RHS)*. Contoh sederhana dari *association rules* adalah sebagai berikut”Jika seseorang membeli palu maka mereka membeli juga paku” *antecedent* disini membeli palu dan *consequent* adalah membeli paku (Two Crows Corporation, 2005). Analisa mengenai adanya keterkaitan dari suatu item dengan item lainnya yang terjadi pada suatu transaksi dari satu keranjang pembelian yang dilakukan algoritma *association rules*, menyebabkan teknik *association rules* disebut juga dengan *market basket analysis*.

Denton dan Beseman, berpendapat bahwa penggunaan *database data mining* yang menerapkan *Relational Database Management System (RDBMS)* akan cukup membantu dalam proses *association rules*. Pendapat ini juga didukung oleh Martine, di mana data yang dihasilkan dapat ditampilkan dengan metode statistik melalui perhitungan (et.al, 2006).

2.5.2 Algoritma Apriori

Algoritma *apriori* adalah algoritma yang paling terkenal untuk menemukan pola frekuensi tinggi (Dunham 2003). Pola frekuensi tinggi adalah pola-pola item di dalam suatu *database* yang memiliki *support* di atas ambang batas tertentu yang disebut dengan istilah *minimum support*. Pola frekuensi tinggi digunakan untuk menyusun *Association Rules* dan juga beberapa teknik *data mining* lainnya. Walaupun akhir-akhir ini dikembangkan banyak algoritma yang lebih efisien dari *apriori* seperti *FP-growth*, LCM dan sebagainya, tetapi *apriori*

tetap menjadi algoritma yang paling banyak digunakan dalam produk komersial untuk *data mining*, karena dianggap algoritma yang paling mapan.

Algoritma *apriori* dibagi menjadi beberapa tahap yang disebut dengan iterasi. Tiap iterasi menghasilkan pola frekuensi tinggi dengan panjang yang sama dimulai dari iterasi pertama yang menghasilkan pola frekuensi tinggi dengan panjang satu. Pada iterasi pertama ini, *support* dari setiap item dihitung dengan men-scan *database*. Setelah *support* dari setiap item didapat, item yang memiliki *support* di atas minimum *support* dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat 1-*itemset*. Singkatan *k-itemset* berarti satu set yang terdiri k item.

Iterasi kedua menghasilkan 2 *itemset* yang tiap setnya memiliki dua item. Pertama dibuat kandidat 2 *itemset* dari kombinasi semua 1-*itemset*. Lalu untuk tiap kandidat 2 *itemset* ini dihitung *support*nya dengan men-scan *database*. *Support* disini artinya jumlah transaksi dalam *database* yang mengandung kedua item set dalam kandidat 2-*itemset*. Setelah *support* dari semua kandidat 2 *itemset* didapatkan, kandidat 2-*itemset* yang memenuhi syarat minimum *support* dapat ditetapkan sebagai 2-*itemset* yang juga merupakan pola frekuensi tinggi dengan panjang 2.

Untuk selanjutnya pada iterasi ke-k dapat dibagi lagi menjadi beberapa bagian:

- a. Pembentukan kandidat *itemset*, Kandidat *k-itemset* dibentuk dari kombinasi (k-1)-*itemset* yang didapat dari iterasi sebelumnya. Satu ciri dari algoritma *Apriori* adalah adanya pemangkasan kandidat *k-itemset* yang subset-nya yang berisi k-1 item tidak termasuk dalam pola frekuensi tinggi dengan panjang k-1.
- b. Penghitungan *support* dari tiap kandidat *k-itemset*. *Support* dari tiap kandidat *k-itemset* didapat dengan men-scan *database* untuk menghitung jumlah transaksi yang memuat semua item di dalam kandidat *k-itemset* tsb. Ini adalah juga ciri dari algoritme *Apriori* dimana diperlukan penghitungan dengan scan seluruh *database* sebanyak *k-itemset* terpanjang.

- c. Tetapkan pola frekuensi tinggi. Pola frekuensi tinggi yang memuat k item atau k -itemset ditetapkan dari kandidat k -itemset yang *support*-nya lebih besar dari minimum *support*.
- d. Bila tidak didapat pola frekuensi tinggi baru maka seluruh proses dihentikan. Bila tidak, maka k ditambah satu dan kembali ke bagian 1.

Pseudocode dari algoritma *Apriori* dapat dilihat di gambar berikut :

```

L1 := { large 1-itemsets };
k := 2; // k represents the pass number
while (Lk-1 ≠ ∅) do
begin
  Ck := New candidates of size k generated from Lk-1; (apriori-gen)
  forall transactions t ∈ D do
    Increment the count of all candidates in Ck that are contained in t;
  Lk := All candidates in Ck with minimum support;
  k := k + 1;
end
Answer := ∪k Lk;

```

Gambar 2.8 *Pseudocode* dari algoritma *Apriori* (R. Agrawal and R. Srikant.1994)

Sedangkan *pseudocode* dari pembentukan kandidat *itemset* bersama pemangkasannya diberikan di Gambar 2.9 berikut:

```

-(1) Join Step
insert into candidate k-itemset
select p.item1, p.item2, ..., p.itemk-1
from large (k - 1)-itemset p, large (k - 1)-itemset q
where p.item1 = q.item1, ..., p.itemk-2 = q.itemk-2, p.itemk-1 < q.itemk-1;

(2) Prune Step
forall itemsets c ∈ candidate k-itemset do
  forall (k - 1)-subsets s of c do
    if (s ∉ large(k - 1)-itemset) then
      delete c from candidate k-itemset;

```

Gambar 2.9 *Pseudocode* pembentukan kandidat *itemset* (R. Agrawal and R. Srikant.1994)

Secara ringkas, langkah-langkah algoritma *apriori* adalah sebagai berikut (indodm) :

- a. *Scan Database*.
- b. Hitung *support* masing-masing *itemset*
- c. *Pruning*, yaitu membuang *itemset* yang tidak memenuhi *support*, yang memenuhi *support* disebut kandidat untuk *itemset* pada scan berikutnya.
- d. Kombinasikan semua kandidat.
- e. Ulangi langkah a-d sampai tidak ditemukan lagi *frequent itemset*.

Sebuah contoh dari penerapan algoritma *apriori* diilustrasikan pada gambar 2.10. Pada contoh tersebut, minimum *support* adalah 50% atau minimal *support*nya adalah 2. Pada iterasi pertama, item yang *support*nya di bawah 2 dieliminasi dari 1 *itemset* L1. Kemudian kandidat 2-*itemset* C2 dari iterasi kedua dibentuk dari *cross product* item-item yang ada di L1. setelah kandidat 2 *itemset* itu dihitung dari *database*. Ditetapkan 2 *itemset* L2. Proses serupa berulang di iterasi ketiga, tetapi perhatikan bahwa selain {2,3,5} yang menjadi kandidat 3 – *itemset* C3 sebenarnya ada juga *itemset* {1,2,3} dan {1,3,5} yang dapat diperoleh dari kombinasi item-item di L2, tetapi kedua *itemset* itu dipangkas karena {2,3} dan {1,5} tidak ada di L2. Proses ini berulang sampai tidak ada lagi kandidat baru yang dapat dihasilkan di iterasi ke 4.

Transaction D			C ₁		L ₁	
TID	Items		Itemset	Count	Itemset	Count
100	1,3,4	Scan D ⇒	{1}	2	{1}	2
200	2,3,5		{2}	3	{2}	3
300	1,2,3,5		{3}	3	{3}	3
400	2,5		{4}	1	{5}	3
			{5}	3		
C ₂			C ₂		L ₂	
Itemset			Itemset	Count	Itemset	Count
{1,2}		Scan D ⇒	{1,2}	1	{1,3}	2
{1,3}			{1,3}	2	{2,3}	2
{1,5}			{1,5}	1	{2,5}	3
{2,3}			{2,3}	2	{3,5}	2
{2,5}			{2,5}	3		
{3,5}			{3,5}	2		
C ₃			C ₃		L ₃	
Itemset			Itemset	Count	Itemset	Count
{2,3,5}		Scan D ⇒	{2,3,5}	2	{2,3,5}	2

Gambar 2.10 Contoh Aplikasi algoritma *Apriori* (R.Agrawal and R. Srikant.1994)

Dalam contoh tersebut dapat dilihat bahwa *apriori* dapat mengurangi jumlah kandidat yang harus dihitung *supportnya* dengan pemangkasan (*pruning*). Misalnya kandidat 3 *itemset* dapat dikurangi dari 3 menjadi 1 saja. Pengurangan jumlah kandidat ini merupakan sebab utama peningkatan performa *Apriori*.

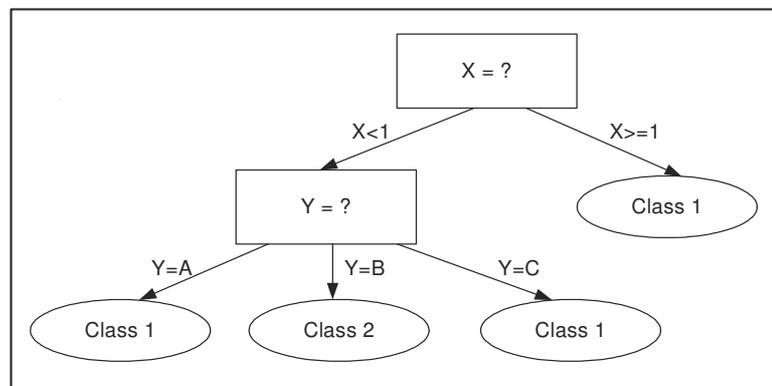
Namun di lain pihak *apriori* memiliki kelemahan karena harus melakukan *scan database* setiap kali iterasi, sehingga waktu yang diperlukan bertambah dengan makin banyak iterasi.

2.5.3 Classification

Classification adalah proses pembelajaran fungsi, dimana bila diberikan suatu input sebagai variabel kedalam barisan data yang dipersentasikan maka akan dapat memetakan (mengklasifikasi) sebuah item data kedalam salah satu dari beberapa kelas yang sudah terdefiniskan (Baricthi, 2004). *Classification* dapat dikatakan sebagai pembelajaran dalam distribut dan *machine learning* (an,2006). Dibawah ini beberapa teknik algoritma *classification*:

- *Decision Tree*

Decision tree adalah salah satu algoritma *classification* yang paling populer. *Decision tree* adalah model prediksi menggunakan struktur *tree*, dimana setiap simpul internal menandakan suatu tes pada *attribute*, setiap cabang merepresentasikan hasil tes dan simpul hasil merupakan dari pemetaan kelas atau distribusi kelas. Contoh sederhana dari *decision tree* dapat dilihat pada gambar 2.11

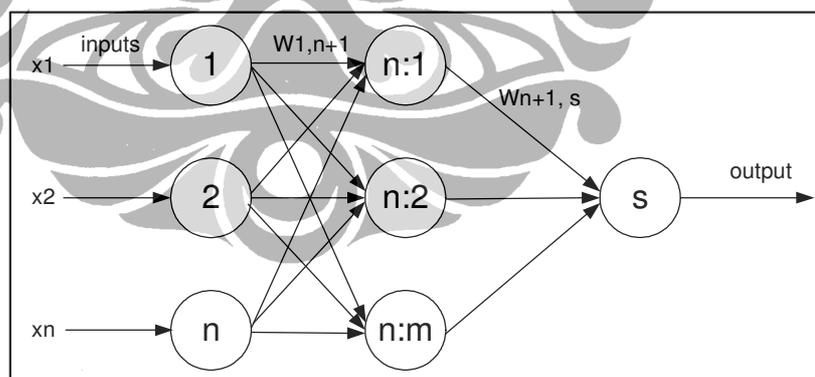


Gambar 2.11 *Decision tree* dengan pengujian pada atribut X dan Y (An, 2006)

Dijelaskan bahwa awal *decision tree* dimulai dengan menginputkan variabel terhadap X, dimana bila pencocokan variabel tersebut lebih besar atau sama dengan satu hasil dari pengujian ini akan terjadi kelas data dua dan nilai yang lebih kecil dari satu masih dapat didefinisikan kembali melalui input variabel terhadap Y, dimana bila hasil pengujian tersebut hasilnya adalah B maka dapat dimasukkan ke dalam kelas data dua, dan bila hasil pengujianya hasilnya A atau C maka akan dimasukkan kedalam kelas satu.

- *Neural Network*

Neural network adalah penggunaan komputasi sederhana untuk menguji data dan mengembangkan model untuk membantu identifikasi pola-pola atau struktur yang bernilai (Smith, 2006). *Neural network* yang merferensi pada *artificial neural network* (Marti, 2006). *Neural network* digunakan untuk mempelajari pola-pola yang ada sebagai referensi dalam memprediksi kejadian di masa depan dan mengklasifikasikan data sebelumnya tidak terlihat kedalam basis *group* yang terdefinisi dari observasi karakteristik berdasarkan informasi yang diperoleh dari sejarah pengujian data. Contoh diagram dari *neural network* ditunjukkan oleh gambar 2.12



Gambar 2.12 *Neural network diagram* (Martin, 2006)

2.5.3 Segmentasi

Segmentasi (pengelompokan) merupakan teknik *data mining* deskripsi yang melakukan proses pengelompokan sejumlah kumpulan pola-pola kedalam bagian

atau *segmen* didasari pada kesesuaian perkiraan kedekatan atau kesamaan terbesar yang ada di pola-pola tersebut. Prinsip dari *segmentasi* adalah memaksimalkan kesamaan yang ada antara anggota satu kelas dan meminimalkan kesamaan antar *segmen* (Two Crows Corporation, 2006), (Baricthi, 2004). Berbeda dengan *association rules* dan *classification* yang telah ditentukan definisi kelas data sebelum proses pemodelan, *segmentasi* tidak didasari kelas data terdahulu dalam melakukan pengelompokan, bahkan dengan *segmen* dapat ditemukan kelas data yang sebelumnya tidak diketahui (Larose,2005).

Penerapan *segmentasi* dalam teknik *data mining*, cukup luas di berbagai bidang bantuan algoritma *segmentasi* (Huang, 2006), dapat dipahami *natural segmen* atau struktur yang terdapat dalam *database*. Penggunaan metode algoritma *segmen* seperti algoritma K-Means (Ghosh,2003), algoritma *center-based segmentasi* (Zhang, 2006), merupakan hal utama dari kegiatan proses teknik *segmentasi*.

2.6 Produk Wireline

Produk *Wireline* adalah Sambungan fasilitas telepon yang menggunakan teknologi *Public Services Telephony Network* (PSTN) dan menggunakan jaringan akses berbasis kabel fisik menggunakan kabel tembaga ataupun kabel *fiber/optic*. Dapat digunakan untuk fungsi telepon, faksimili atau data/internet dengan penambahan modem oleh pelanggan. Kemudian untuk layanan suara/voice terdiri dari percakapan Lokal, Sambungan Langsung Jarak Jauh (*SLJJ*) dan sambungan langsung Internasional (*SLI*).

Adapun Fasilitas layanan sebagai pelengkap / *supplementary service* (fitur) yang ditawarkan bagi pelanggan *Wireline* meliputi (Keputusan Direksi PT.XYZ, 2006):

- *Call Waiting* : untuk dapat mengetahui dan menerima panggilan telepon tanpa memutus hubungan telepon yang sedang berjalan.
- *Call Forwarding* : untuk mengalihkan tujuan panggilan ke nomor yang diinginkan pelanggan.

- *Hotline* : untuk menyambungkan hubungan telepon ke suatu nomor tertentu tanpa perlu dilakukan dialing oleh pelanggan.
- *Three Party* : untuk melakukan pembicaraan telepon dari 3 (tiga) pengguna/nomor yang berbeda.
- *Abbreviated Dialing* : untuk memudahkan dialing dengan memberikan nomor sandi untuk beberapa nomor telepon tujuan.
- *Hunting* : untuk memudahkan pengingatan nomor telepon dengan pemberian nomor panggil untuk beberapa nomor telepon yang dikelompokkan.
- *Keyword* : untuk melakukan klasifikasi kemampuan telepon (incoming only, local only atau SLJJ) sesuai keinginan pelanggan dengan melakukan proses setting tertentu melalui terminal telepon.
- *Do Not Disturb* : untuk menolak panggilan telepon pada saat pelanggan tidak ingin dihubungi.
- *Wake Up Call* : untuk melakukan panggilan sebagai announcement pada waktu tertentu sesuai keinginan pelanggan.
- *CLIP – Calling Line Identification Presentation* : untuk mengetahui nomor pemanggil secara langsung melalui terminal khusus.
- *Voice Mail Box* : Fasilitas layanan yang diberikan bagi pelanggan telepon yang berguna untuk merekam pesan di saat telepon sedang digunakan, tidak ada yang menjawab atau pada saat telepon mengalami kerusakan. Pelanggan akan mendapat notifikasi apabila ada pesan di dalam mailboxnya.
- *TeleInfo* : layanan jasa informasi dan hiburan yang dikemas secara interaktif untuk pelanggan dengan voice response system (VRS). Setiap wilayah pelayanan TELKOM menyediakan jasa informasi dan hiburan yang berbeda.
- *SMS* : layanan jasa pengiriman pesan dengan menggunakan media data dimana pelanggan dapat mengirim dan menerima pesan secara tertulis, dan dapat dikirim/diterima dalam waktu yang singkat.

2.7 Loyalitas Pelanggan

Pengukuran terhadap kepuasan maupun ketidakpuasan pelanggan pada merek merupakan indikator yang penting dari loyalitas merek. Bila ketidakpuasan pelanggan terhadap suatu merek rendah, maka pada umumnya tidak cukup alasan bagi pelanggan untuk beralih mengkonsumsi merek lain kecuali bila ada faktor-faktor penarik yang sangat kuat.

Pelanggan yang puas akan melakukan transaksi bisnis lebih banyak dan lebih sering. Mereka juga merekomendasikan perusahaan atau merek kepada kerabat dan teman-temannya. Menurut Gerson, 2001, bahwa, hubungan antara penjualan, pelayanan, kepuasan dan profit bersifat langsung. Semakin puas seorang pelanggan, semakin banyak uang yang dibelanjakannya. Variabel laten loyal merupakan bentuk dari perilaku konsumen, khususnya sebagai variabel yang menjelaskan tentang niat atau minat untuk berbuat di masa datang (Behavior Intention). Pengukuran terhadap variabel loyalitas (loyal) ini dilakukan dengan pendekatan sikap dan perilaku.

Pengukuran Sikap (*Attitudinal Measurements*), dilakukan terhadap konsistensi pembelian dan pembelian ulang. Seperti yang dikemukakan oleh Morgan dan Hunt, 1998; Hennig-Thurau and Klee, 1997; Moorman, Zaltman, and Deshpande, dalam Seyhmus baloglu, 2002, bahwa konstruk kunci dari *relationship marketing* adalah hubungan antara kepercayaan dengan komitmen. Komitmen konsumen untuk tetap berhubungan dengan perusahaan karena adanya keyakinan atau kepercayaan kepada perusahaan sehingga akan melakukan pembelian ulang.

Loyalitas terhadap produk/jasa perusahaan (merek) didefinisikan sebagai sikap menyenangkan (*favorable*) terhadap sesuatu merek, yang direpresentasikan dalam pembelian yang konsisten terhadap merek itu sepanjang waktu. Kedua pengukuran tersebut merupakan hasil dari proses psychological or emotional attachment. Di sisi lain, Gale (1997) mengatakan bahwa tingkat loyalitas pelanggan dipengaruhi oleh persepsi konsumen terhadap nilai yang ditawarkan oleh perusahaan. Hasil dari sebuah evaluasi konsumen individual, terhadap suatu produk atau jasa dapat memenuhi harapannya atau mampu memuaskannya, maka di masa datang akan terjadi pembelian ulang.

Sehingga dari uraian diatas loyalitas pelanggan dapat diukur dari pola-pola perilaku sebagai berikut :

- Melakukan penggunaan produk secara berulang dan teratur
- Menggunakan produk antar lini
- Mereferensikan kepada orang lain
- Menunjukkan kekebalan terhadap tarikan dari pesaing

Hal-hal tersebut diatas dapat dijadikan ukuran apakah konsumen telah menjadi pelanggan yang loyal atau tidak.

