BAB 2
TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 *Business Intelligence*

Business intelligence (BI) adalah sebuah proses analisis yang mengubah data internal dan data eksternal menjadi informasi tentang kapabilitas, posisi pasar, aktifitas dan tujuan yang harus dicapai oleh perusahaan untuk tetap kompetitif. BI berdiri berdasarkan konsep sistem informasi seperti *Online Analytical Processing* (OLAP), pembuatan *query*, pelaporan, dan *data mining* yang menyediakan metode-metode berbeda untuk sebuah analisis data bisnis (J. Schiefer, 2005). BI bertujuan untuk menyajikan berbagai informasi yang disesuaikan dengan kebutuhan setiap penggunanya. Informasi tersebut bisa berasal dari mana saja, kemudian data tersebut diolah dan disajikan dalam bentuk informasi yang dapat dimengerti oleh penggunanya untuk memenuhi tujuan perusahaan.

Terdapat beberapa tantangan dalam melakukan pengembangan *business intelligence* diantaranya (Grigori, 2004) :

1. Mengidentifikasi arsitektur dan teknologi yang dapat menghasilkan fungsi dari BPI dan memahami bagaimana mengaplikasikan atau memodifikasi teknologi ini untuk mencapat tujuan perusahaan.
2. Dapat mendefinisikan konsep dan metriks yang memungkinkan analisis proses di tingkat bisnis dan kualitatif.
3. Mengembangkan teknik untuk memudahkan bagi analis dalam menggunakan aplikasi dan mendapatkan pengetahuan yang mereka butuhkan bahkan apabila memungkinkan tanpa perlu menuliskan perintah pemrograman apapun.

4. Memahami bagaimana interaksi antara *business process management systems* (BPMS) dan pengguna untuk menghasilkan laporan dan memperbaiki situasi kritikal.

Terdapat beberapa bagian dalam solusi *business intelligence,* yaitu keseluruhan proses dalam *business intelligence* dapat diterjemahkan menjadi langkah-langkah dibawah ini :

a. Identifikasi masalah bisnis yang perlu diselesaikan dengan gudang data dan menentukan data yang diperlukan untuk menyelesaikan masalah tersebut.
b. Identifikasi lokasi dari data yang diperlukan dan mengambilnya dari sumber penyimpanan.
c. Mengubah data yang diperoleh dari beragam sumber tersebut ke dalam sebuah data yang konsisten.
d. Mengambil data yang telah dirubah tersebut ke dalam lokasi yang tersentralisasi.
e. Membuat sebuah gudang data dengan data yang ada dalam lokasi yang tersentralisasi tersebut.
f. Memasang sebuah produk atau aplikasi yang dapat memberikan akses ke data yang ada dalam *cube* tadi. Ada berbagai macam jalan dan cara untuk berbagai macam tipe pekerjaan ketika berurusan dengan *cube*.

2.1.2 Arsitektur *Business Intelligence*

Arsitektur dari sistem *Business Intelligence* terdiri atas enam komponen utama, yaitu (Vercellis, 2009) :

1. *Data Source*

Tahapan *data source* bertugas untuk mengumpulkan dan mengintegrasikan data yang disimpan dalam berbagai sumber yang beragam dan berbeda baik asal data tersebut dan jenisnya. Sumber data ini berasal dari data traksaksional pada

operational system, tetapi dapat juga berasal dari dokumen yang tidak terstruktur seperti *email* dan data yang dikirimkan oleh pihak luar.

2. *Data Warehouse*

Tahapan *data warehouse* menggunakan *extraction* dan *transformation tool* yang dikenal sebagai proses ETL (*Extract, Transform, Loading*), data yang berasal dari berbagai sumber yang berbeda disimpan ke dalam basis data yang ditunjukkan untuk mendukung proses analisis *business intelligence.*

3. *Data Exploration*

Pada tahapan 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 kepututsan berdasarkan hipotesis mereka seniri 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 learning*) dan teknik *data mining.*

5. *Optimiztion*

Pada tahap ini akan menghasilkan solusi, solusi terbaik akan dipilih dari sekian solusi 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 makin process.* Walaupun metodologi *business intelligence* berhasi diterapkan,pilihan untuk mengambil sebuah keputusan tetap ada ditangan para pengambil keputusan tersebut.

2.1.3 Jenis *Business Intelligence*

Terdapat lima jenis atau kategori dari *business intelligence*, yaitu :

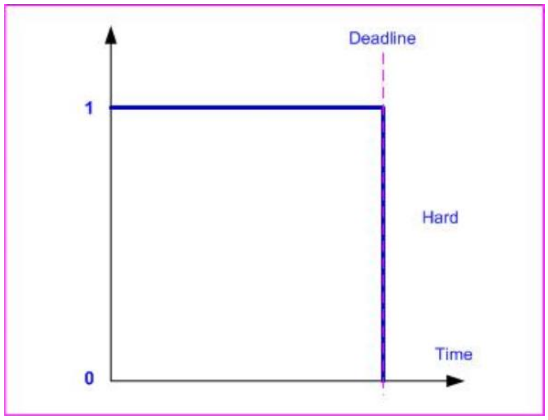
1. *Enterprise Reporting,* digunakan untuk menghasilkan laporan-laporan statis yang di distribusikan ke banyak orang. Jenis laporan ini sangat sesuai untuk laporan operasional dan *dashboard.*
2. *Cube Analysis,* digunakan untuk menyediakan analisis OLTP multidimensional yang ditujukan untuk manajer bisnis dalam lingkungan terbatas.
3. *Ad-Hoc Query and Analysis,* digunakan untuk memberikan akses kepada pengguna agar dapat melakukan *query* pada basis data, dan menggali informasi sampai pada tingkat paling dasar dari informasi transaksional. *Query* ini berfungsi untuk menjelaja informasi yang dilakukan oleh pengguna.
4. *Statistical Analysis and Data Mining,* digunakan untuk melakukan analisis prediksi atau menentukan korelasi sebab akibat diantara dua matrik.
5. *Delivery Report and Alert,* digunakan secara proaktif untuk mengirimkan laporan secara lengkap atau memberikan peringatan kepada populasi pengguna yang besar atau banyak.

Pada penelitian ini digunakan jenis atau kategori *Statistial Analysis and Data Mining.* Proses yang digunakan untuk menganalisa dan menjelajahi informasi mengenai prediksi delay kendaraan berdasarkan korelasi antara jadwal keberangkatan, delay tiba, delay berangkat, dan tujuan.

2.1.4 *Real-time System*

Real-time system (sistem waktu nyata) harus menghasilkan respon yang tepat dalam batas waktu yang telah ditentukkan. Jika respon melewati batas waktu tersebut, maka terjadi degradasi performansi atau kegagalan sistem. Sebuah sistem *real-time* adalah sistem yang kebenarannya secara logis didasarkan pada kebenaran hasil keluaran sistem dan ketepatan waktu hasil tersebut saat dikeluarkan (Willey, 2004). menurut fungsinya sistem ini terbagi menjadi *hard real-time* dan *soft real-time.*

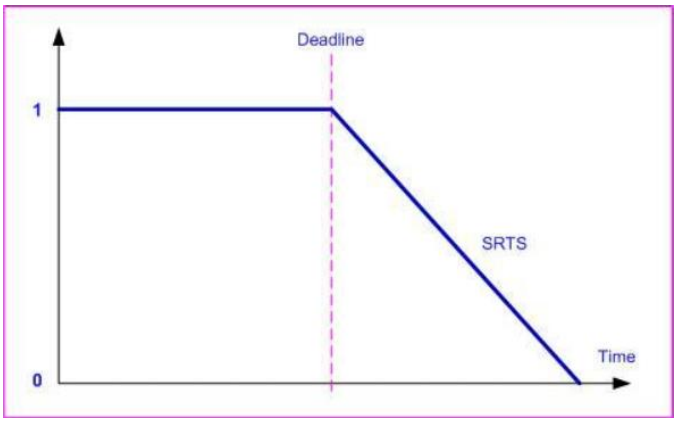
Sistem *hard real-time* dibutuhkan untuk menyelesaikan *critical task* dengan jaminan waktu tertentu. Jika kebutuhan waktu tidak terpenuhi, maka aplikasi akan gagal. Secara umum, sebuah proses di kirim dengan sebuah pernyataan jumlah waktu yang dibutuhkan untuk menyelesaikan atau menjalankan proses *input / output.* Contoh pada kehidupan sehari-hari adalah sistem pengontrol pesawat. Dalam hal ini keterlambatan tidak boleh terjadi, karena dapat berakibat tidak terkontrolnya pesawat. Grafik *hard real-time* dapat dilihat pada Gambar 2.1



Gambar 2.1 Grafik Sistem *Hard Real-time* **(Siyamta)**

Sistem *soft real-time* memiliki sedikit kelonggaran waktu dibandingkan dengan sistem *hard real-time,* karena perangkat lunak hanya mendapatkan prioritas dalam menjalankan prosesnya, sehingga risiko kegagalan tidak berakibat fatal. Adanya keterlambatan waktu respon akan mengakibatkan penurunan kinerja dan mengakibatkan waktu tunda yang lebih lama. Contoh pada kehidupan sehari-

hari adalah pada alat penjual otomatis. Jika mesin yang menggunakan sistem ini telah lama digunakan, maka mesin tersebut dapat mengalami penurunan kualitas. Keterlambatan pada sistem ini tidak menyebabkan kecelakaan atau akibat fatal lainnya. grafik sistem *soft real-time* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Grafik Sistem *Soft Real-time* **(Siyamta)**

Setelah batas waktu yang diberikan telah habis, pada sistem *hard real-time*, sistem yang dijalankan akan langsung dihentikan. Akan tetapi, pada sistem *soft real-time*, sistem yang telah habis masa waktu pengerjaan tugasnya, dihentikan secara bertahap atau masih diberikan toleransi waktu.

Pada penelitian ini menggunakan sistem *soft real-time*. Prioritas pemrosesan pada penelitian ini adalah saat mencari kombinasi antar atribut pada algoritma *apriori*.

2.1.5 *Data Mining*

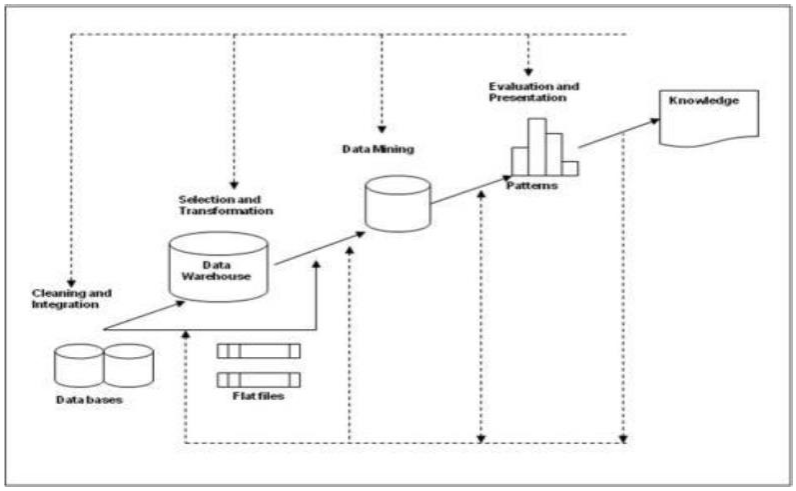
Data mining merupakan proses untuk menggali nilai tambah, berupa informasi yang tidak diketahui dari suatu basis data. Proses yang dilakukan pada *data mining* adalah dengan melakukan penggalian pola-pola tertentu dari suatu basis data, yang bertujuan untuk memanipulasi data menjadi sebuah informasi

yang lebih berharga, yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data.

Data mining juga dikenal dengan *Knowledge Discovery in Database* (KDD). Secara umum, definisi *data mining* dapat diartikan sebagai berikut (J. Han, 2006):

1. Merupakan proses untuk menemukan pola yang menarik dari data yang tersimpan pada basis data dalam jumlah besar.
2. Ekstraksi menjadi suatu informasi yang berguna atau menarik pola yang sebelumnya tidak/belum diketahui potensial kegunaannya atau pengetahan dari data yang di simpan pada basis data dalam jumlah besar.
3. Ekploirasi dari analisa secara otomatis atau semiotomatis terhadap data dalam jumlah besar untuk mencari suatu pola dan aturan yang memiliki suatu arti tertentu.

Data mining sangat diperlukan, terutama untuk mengelola data yang sangat besar yang dapat memudahkan aktifitas *recording* suatu transaksi dan untuk proses *data warehousing* agar dapat memberikan informasi yang akurat bagi pengguna. Alasan utama *data mining* dibutuhkan dalam industi informasi adalah karena tersedianya data dalam jumlah besar dan semakin besarnya kebutuhan untuk menguba data tersebut menjadi informasi dan pengetahuan yang berguna. Sesuai fokus pada bidang ilmu ini yaitu untuk melakukan kegiatan ektraksi atau menggali pengetahuan dari data yang berukuran besar menjadi sebuah informasi yang nantinya akan berguna untuk pengembangan. Berikut merupakan langkah-langkah dalam melakukan *data mining.*



Gambar 2.3 Konsep *Data Mining* **(J. Han, 2006)**

Penjelasan dari konsep *data mining* yang dijelaskan pada Gambar 2.3 adalah sebagai berikut (J. Han, 2006):

1. *Data cleaning*, yaitu proses untuk menghilangkan *noise* data yang tidak koniten.
2. *Data integration,* yaitu proses penggabungan beberapa *file* atau basis data.
3. *Data selection*, yaitu data yang sudah relevan untuk tugas analisis akan dikembalikan kembali ke basis data untuk proses *data mining.*
4. *Data transformation,* yaitu data berubah satu persatu menjadi bentuk yang tepat untuk menambang dengan ringkasan performa atau operasi agresi.
5. *Data mining*, yaitu metode yang intelijen yang digunakan untuk mengekstrak pola pada data.
6. *Knowledge discovery*, yaitu metode yang intelijen untuk mengekstrak pola pada data untuk mendapatkan pengetahuan.
7. *Patern evolution,* yaitu untuk mengidentifikasi pola yang benar-benar menarik, yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik.

8. *Knowledge presentation,* yaitu gambaran teknik visualisasi dan pengetahuan yang digunakan untuk memberikan pengetahuan yang telah ditambahkan kepada pengguna.

2.1.6 *Data Preparation*

Data Preparation atau bisa disebut juga *data preprocessing* merupakan suatu proses atau langkah yang dilakukan untuk membuat data mentah menjadi data yang berkualitas yang bisa menjadi masukkan yang baik untuk melakukan proses data mining. Dalam prakteknya, ditemukan bahwa data *cleaning* dan *preparation* membutuhkan total usaha 80% dari usaha untuk merekayasa data, sehingga menjadikan data *preparation* sebagai proes yang krusial (Mannewongvatana, 2010).

Tahapan data *preprocessing* diperlukan untuk melakukan *data mining*, hal ini disebabkan oleh beberapa faktor, diantaranya (R. J. A. Little, 2002) :

1. Data *real world* merupakan data kotor.

Data *real world* dapat mengandung data yang tidak lengkap, terdapat *noise,* tidak konsisten, yang dikarenakan :

a. Tidak lengkap (*incomplete*), yaitu kekurangan nilai atribut atau hanya mengandung agregat data (contoh : alamat = “ ”).
b. *Noise,* yaitu masih mengandung *error* yang dikarenakan nilai yang tidak wajar. (contoh umur = -10).
c. Tidak konsisten (*inconsistent*), yaitu data yang mengandung discrepansi dalam *code* dan nama atau singkatnya data yang ada tidak konsisten (contoh : dulu rating = 1,2,3 sekarang a,b,c).

2. Sistem *mining* dengan performa tinggi membutuhkan data yang berkualitas.

Data preprocessing menghasilkan dataset yang lebih sedikit daripada dataset yang asli, ini dapat meningkatkan efisiensi dari *data mining*. Langkah ini mengandung :

a. Memilih data yang relevan dengan memilih atribut, menghilangkan anomali, menghilangkan data duplikat.
b. Mengurangi data menggunakan *sampling*

3. Data yang berkualitas menghasilkan pola yang berkualitas.

Dengan *data preprocessing,* maka data yang dihasilkan adalah data yang berkualitas, yang mengarah pada pla yang berkualitas pula, dengan :

a. Mengembalikan data yang tidak lengkap
b. Membenarkan *error,* atau menghilangkan *outliers.*
c. Membenahi data yang bertentangan.

Bila data yang dihasilkan melalui proses ini berupa data yang berkualitas, maka roses *mining* untuk menemukan *hidden knowledge* akan dipermudah, karena hasil *mining* nantinya tidak akan menyimpang dari data yang ada, lebih efisien dan berkualitas, dikarenakan tidak adanya permasalahan pada data, misalnya *missing data* (Mannewongvatana, 2010).

Tahapan yang diperlukan dalam melakukan *data preprocessing* adalah sebagai berikut :

1. *Data cleaning,* mengisi / mengganti nilai-nilai yang hilang, menghaluskan data yang *noisy,* mengidentifikasi dan menghilangkan data yang tidak wajar, dan menyelesaikan masalah *inconsistent data.*
2. *Data integration,* menggabungkan beberapa basis data dan atau *file* menjadi satu sehingga didapatkan sumber data yang besar.
3. *Data transformation,* melakukan normalisasi dan agregasi data.
4. *Data reduction,* mengurangi volume data, namun tetap mempertahankan arti dalam hal hasil analisis data.
5. *Data discretization,* merupkan bagian dari data *reduction* dengan memperhitungkan data yang signifikan, khususnya pada data numerik.

2.1.7 *Assosiation Rule*

Assosiation rule merupakan salah satu teknik utama dalam *market basket analysis.* Teknik ini digunakan untuk mencari hubungan antar *item* dalam suatu *dataset* dan menampilkan bentuk *assosiation rule* (Santoni, 2012). *Assosiation rule* akan menemuka pola tertentu untuk mengasosiasikan antar data. Untuk mencari *association rule* dari suatu kumppulan data, tahap pertama yang dilakukan adalah mencari *frequent itemset* terlebih dahulu. *Frequent itemset* merupakan sekumpulan *item* yang sering muncul secara bersamaan. Setelah semua pola *frequent itemset* ditemukan, langkah selanjutnya adalah mencari aturan asosiatif atau keterkaitan yang memnuhi syarat yang telah ditentukan. Contoh aturan asosiatif adalah apabila pelanggan melakukan perjalanan dari bandung menuju Jakarta pada pukul 06.00 maka akan mengalami kemacetan, untuk aturan asosiatifnya ditunjukkan sebagai berikut :

$$Bandung, Jakarta, 06.00 \rightarrow Macet\ [support = 2\%, confidence = 60\%]$$

Assosiation rule memerlukan suatu variabel ukuran yang ditentukan sendiri oleh pengguna. Variabel tersebut digunakan untuk menentukan batasan sejauh mana atau sebanyak apa *output* yang diinginkan pengguna. *Support* dan *confidence* adalah sebuah ukuran kepercayaan dan kegunaan suatu pola yang telah ditemukan. Seperti di dalam contoh di atas nilai *support* 2% menunjukkan bahwa keseluruhan dari total transaksi yang terjadi antara konsumen yang melakukan perjalanan ke Bandung menuju Jakarta atau sebaliknya, dan melakukan perjalanan pada pukul 06.00 adalah sebanyak 2%. Sedangkan nilai *confidence* 60% menunjukkan bahwa konsumen yang melakukan perjalanan ke Bandung menuju Jakarta dan melakukan perjalanan pada pukul 06.00 pasti mengalami kemacetan sebanyak 60%.

2.1.8 Algoritma Apriori

Algoritma *apriori* merupakan salah satu algoritma yang digunakan untuk melakukan pencarian jumlah kemunculan atribut. Algoritma ini menggunakan pengetahuan kemunculan suatu atribut yang telah diketahui sebelumnya, untuk melakukan penggalian informasi pada proses selanjutnya. Pada algoritma *apriori*, menentukan kandidat yang mungkin muncul dengan cara memperhatikan *minimum support* dan *minimum confidence.* Nilai *support* adalah nilai pengunjung atau persentase kombinsi atribut dalam basis data.

Untuk mendapatkan nilai *support* untuk sebuah atribut, digunakan rumus sebagai berikut :

$$Support\ (A) = \left(\frac{Jumlah\ transaksi\ mengandung\ A}{Total\ Transaksi}\right) x\ 100\% \dots\dots\dots\dots (2.1)$$

Sedangkan untuk mencari nilai *support* untuk dua *item*, digunakan rumus sebagai berikut :

$$Support\ (A, B) = \left(\frac{Jumlah\ transaksi\ mengandung\ A\ dan\ B}{Total\ Transaksi}\right) x\ 100\%.. (2.2)$$

Sedangkan nilai *confidence* adalah nilai kepastian, yaitu kuatnya hubungan antar *item.* Nilai *confidence* dapat dicari setelah pola frekuensi munculnya sebuah *item* ditemukan. Rumus yang digunakan untuk mencari nilai *confidence* adalah sebagai berikut :

$$Confidence = \frac{Jumlah\ transaksi\ yang\ mengandung\ A\ dan\ B}{Jumlah\ transaksi\ yang\ mengandung\ A} \dots\dots\dots (2.3)$$

Proses utama yang dilakukan dalam algoritma *apriori* untuk mendapatkan *frequent itemset* yaitu :

1. *Join* (Penggabungan)
   Proses ini dilakukan dengan cara mengkombinasikan suatu *item* dengan *item* yang lainnya hingga tidak dapat terbentuk kombinasi lagi.
2. *Prune* (Pemangkasan)

Hasil dari *item* yang telah dikombinasikan, kemudian dipangkas dengan menggunakan *minimum support* yang telah ditentukan.

item A dan *item* B benar saling terkait dan muncul atau dibeli secara bersamaan.

2.1.9 *Change Data Capture* (CDC)

CDC adalah pendekatan yang digunakan untuk integrasi, identifikasi, menangkap, dan mengirimkan perubahan yang dibuat pada sumber data dengan memproses hanya perubahan yang terjadi. Dengan menggunakan CDC, proses integrasi data lebih efisien dan mengurangi biaya dengan mengurangi latensi. (Attunity, 2006).

Terdapat dua model skenario CDC yaitu (Corporation, 2005) :

1. CDC *Pull* : model yang secara periodik meminta perubahan data, setiap waktu menerima sekumpulan *record* yang merepresentasikan semua perubahan yang ditangkap sejak siklus permintaan terakhir. Metode ini hanya menangkap dan memindahkan data yang berubah saja.
2. CDC *Push* : mekanisme pengiriman perubahan data segera setelah perubahan terjadi. Metode ini menggunakan *listeners* yang menunggu *event* perubahan dan *publisher* yang digunakan untuk mengirim dan memberikan notifikasi perubahan secara *real-time.*

Pada penelitian ini menggunakan metode CDC *push*, setiap kegiatan *insert*, *update*, atau *delete* yang terjadi pada tabel sumber akan dipindahkan ke tabel proses CDC yang tepat.

2.2 Penelitian Terkait

Pada penelitian yang dilakukan oleh stephanie pamela adithama pemanfaatan real-time business intelligence digunakan untuk membantu kegiatan akademis pada universitas. Aplikasi yang dibuat digunakan untuk memberikan informasi secara cepat untuk data transaksional mahasiswa, dosen dan pihak yang terkait dengan universitas. Aplikasi ini dapat membantu dalam pengisian borang yang akan digunakan oleh pihak universitas untuk mendapatkan informasi kegiatan secara real-time. Integrasi yang digunakan pada penelitian tersebut menggunakan data warehouse dan CDC (Change Data Capture). Pada penelitian yang dilakukan menggunkan dua model skenario CDC, yaitu CDC *Pull* yang secara periodik meminta perubahan data, setiap waktu menerima sekumpulan *record* yang merepresentasikan semua perubahan yang ditangkap sejak siklus permintaan terakir. Skenario ini mirip dengan proses ETL tradisional, perbedaannya, skenario ini hanya menangkap dan memindahkan hanya data yang berubah saja. CDC *Push* yang secara segera mengirimkan perbahan data ke *tool* ETL setelah perubahan terjadi (Stephanie Pamela Adithama, 2013).

Metode yang di menjadi referensi untuk penelitian ini adalah metode CDC *Push*, proses tersebut dapat diimplementasikan pada tahap *preprocessing* pada *data mining.* Proses yang dapat diimplementasikan saat melakukan proses perubahan terhadap nilai yang akan dimasukkan ke dalam *dataset* penjadwalan.

Penelitian yang dilakukan faizal hendyansyah khrisdian mengenai aturan asosiasi yang digunakan untuk membantu memprediksi delay penerbangan. Penelitian tersebut menggunakan algoritma *apriori* untuk mencari kombinasi penjadwalan yang ada. Pada penelitian tersebut langkah pengujian untuk mendapatkan kombinasi yang baik menggunakan metode ekperimental dengan memberikan berbagai nilai *support*, nilai *confidence*, dan pembatasan terhadap waktu proses. Nilai support yang ditentukan adalah 0, 5, dan 7 serta nilai *confidence* yang ditentukan bernilai 0% - 70%. pada penelitian tersebut diketahui pula bahwa kombinasi yang terbentuk memiliki keakurasian 70%-82% dengan waktu proses tercepat adalah satu detik. Pada penelitian tersebut. Pada penelitian

tersebut didapatkan kekurangan apabila data yang akan diolah banyak, karena nilai support yang sudah ditentukan, sehingga terdapat kemungkinan seluruh jadwal akan masuk ke dalam aturan asosiasi karena sesuai dengan minimum support yang ditentukkan (Khrisdian, 2015).

Pada penelitian yang dilakukan faizal, dalam melakukan *data mining* dataset yang digunakan adalah basis data operasional, dan berdasarkan hasil penelitian yang dilakukan ditemukan saran untuk menguji dengan jumlah data yang lebih besar. Pada penelitian yang dilakukan, atribut yang akan digunakan untuk *data mining* berasal dari satu tabel, sehingga proses *mining* dapat dilakukan dengan cepat karena tabel yang terlibat sedikit, sedangkan pada penelitian yang dilakukan di PT. Citra Tiara Global diketahui bahwa daftar atribut yang diperlukan berasal dari beberapa tabel relasi, sehingga apabila proses *data mining* dilakukan, akan membutuhkan waktu lebih lama dalam proses pengambilan dan pemilihan data yang akan digunakan.

Berdasarkan penelitian terkait, pada penelitian ini akan menggunakan teknik CDC (Change Data Capture) dengan metode *push* untuk melakukan pengambilan data pada tabel-tabel yang terlibat dalam penjadwalan yang nantinya akan dipindah ke dalam sebuah basis data yang akan digunakan untuk proses *data mining.* Proses CDC yang akan digunakan berdasarkan perubahan yang terjadi, berdasarkan fakta di lapangan diketahui bahwa perubahan data penjadwalan memiliki interval 15 menit untuk setiap keberangkatannya, sehingga dengan teknik CDC *Push* dapat membantu untuk mendapatkan update data terbaru secara cepat. Dalam melakukan pencarian kombinasi untuk prediksi penjadwalan akan menggunakan metode *association rule* menggunakan algoritma *apriori* dengan *data stream mining. Data stream mining* digunakan untuk membatasi waktu proses dalam pencarian kombinasi, yang akan disesuaikan berdasarkan kebutuhan dari perusahaan.