

1. Pendahuluan

1.1 Latar Belakang

Kebutuhan akan pengolahan data dari berbagai sumber yang terintegrasi untuk kemudian dijadikan sandaran analisa semakin besar. *Data Warehouse* merupakan koleksi data yang mempunyai sifat berorientasi subjek, terintegrasi, *time-variant*, dan bersifat tetap dari koleksi data dalam mendukung proses pengambilan keputusan [11]. *Data Warehouse* dapat dimanfaatkan untuk kebutuhan tersebut yang memang penggunaannya sudah cukup banyak pada berbagai organisasi maupun perusahaan. Terlebih lagi, *Data Warehouse* dirancang dengan memisahkan beban kerja analisa dari beban kerja transaksi, sehingga penggunaan *Data Warehouse* sebagai acuan dalam pengambilan keputusan menjadi semakin baik. Namun, pengaksesan data pada *Data Warehouse* yang valid dan terkini juga menjadi tuntutan untuk memenuhi berbagai tujuan.

Proses *refreshment* yang berfungsi untuk meng-*update* konten pada *Data Warehouse* berkaitan erat dengan ETL (*Extraction, Transformation, dan Loading*). Untuk membantu ETL, terdapat beberapa metode *capture* perubahan pada *source database* yang kemudian hasilnya dapat digunakan dalam keperluan ETL berikutnya. Sebelumnya, *refreshment* biasa dilakukan sesuai periodik tertentu (misal pada malam hari) ketika sistem sedang tidak digunakan, karena proses ekstraksi dari *source system* terkadang membutuhkan waktu cukup lama sehingga dikhawatirkan akan mengganggu transaksi yang sedang berjalan. Oleh karena itu, *on-time data warehousing* sulit dicapai. Kemudian muncullah teknik CDC (*Change Data Capture*) yang dapat menjanjikan *on-time* (atau biasa disebut *near real-time*) *data warehousing* karena CDC hanya menangkap data yang berubah saja melalui *redo log*, di mana banyaknya data yang diekstraksi dapat secara signifikan direduksi dan berpengaruh besar pada peningkatan efisiensi waktu.

CDC memiliki 2 cara dalam menangkap dan menampilkan perubahan data, yaitu *Synchronous* dan *Asynchronous*. *Synchronous* CDC dapat beresiko terhadap *source system*, tetapi tidak untuk *Asynchronous* CDC. *Asynchronous* CDC sendiri memiliki beberapa metode, salah satunya adalah *HotLog*. Penulis tertarik untuk melakukan implementasi dan analisa terhadap *Asynchronous HotLog* CDC baik dalam segi performansi maupun efisiensi dan efektivitasnya bila diterapkan untuk membantu proses *refreshment* pada *Data Warehouse*.

1.2 Perumusan masalah

Mengacu pada latar belakang di atas, berikut adalah perumusan masalah pada tugas akhir ini:

1. Bagaimana proses dan implementasi *Asynchronous HotLog* CDC dalam mengidentifikasi dan menangkap perubahan pada *source database*.

2. Faktor apa saja yang mempengaruhi proses dan performansi *Asynchronous HotLog CDC*.

1.3 Tujuan

Tujuan dari tugas akhir ini yaitu:

1. Melakukan implementasi *Asynchronous HotLog CDC*.
2. Melakukan analisa faktor-faktor apa saja yang mempengaruhi proses dan performansi *Asynchronous HotLog CDC*, yaitu:
 - a. pendistribusian perubahan data terhadap lebih dari 1 *change set*.
 - b. ukuran *redo log file group* yang berbeda.
 - c. proses *recovery Asynchronous HotLog CDC* ketika terjadi *failure*.

1.4 Batasan Masalah

Untuk mengantisipasi melebarnya penyusunan tugas akhir ini, maka penulis membuat batasan masalah dan ruang lingkup sebagai berikut:

1. Menggunakan Oracle 11g dalam implementasi *Asynchronous HotLog CDC*.
2. Fokus penyusunan tugas akhir ini adalah melakukan implementasi dan analisa performansi *Asynchronous HotLog CDC* yang membantu ETL dalam proses *refreshment*, bukan pada *Data Warehouse* secara global maupun pembangunannya.
3. *Source database* dibangun pada sebuah *standalone personal computer*.
4. Tidak membahas *capture* perubahan data akibat operasi DDL, melainkan hanya membahas *capture* perubahan data akibat operasi DML setelah dilakukan *commit*.

1.5 Metodologi penyelesaian masalah

Penulis menggunakan metodologi berikut untuk menyelesaikan tugas akhir ini:

1. Studi literatur
Tahap awal dari penyusunan tugas akhir ini yaitu melakukan studi literatur dari berbagai sumber seperti paper, buku, jurnal, internet, dan lain-lain dalam menggali informasi mengenai *Data Warehouse*, konsep ETL dan *refreshment*, konsep *near real-time data warehousing*, konsep *Asynchronous HotLog CDC*, implementasi CDC khususnya dengan *Asynchronous HotLog* pada Oracle 11g, dan hal-hal lain yang mendukung dalam penyusunan tugas akhir ini.
2. Perancangan Sistem
Tahap ini berisi perancangan sistem mulai dari skema *source database*, gambaran umum sistem, komponen *publisher*, dan komponen *subscriber*.
3. Implementasi
Bagian ini menjelaskan tahapan-tahapan implementasi *Asynchronous HotLog CDC* sesuai dengan perancangan sistem yang sebelumnya sudah dibuat.
4. Pengujian
Dari hasil implementasi, dilakukan pengujian dengan beberapa skenario uji.

5. Analisa
Untuk semua tahapan implementasi HotLog CDC dan hasil dari pengujian, diambil analisisnya secara mendetail.
6. Pembuatan laporan
Merupakan dokumentasi dari hasil keseluruhan poin-poin di atas.