

SISTEM PERTUKARAN DATA ANTAR BASIS DATA DENGAN XML

Robertus Setiawan Aji Nugroho

Program Studi Ilmu Komputer, Unika Soegijapranata
Jl. Pawiyatan Luhur IV/1 Bendhan Dhuwur Semarang 50234
E-mail: wawan@unika.ac.id

Abstrak

Perkembangan Teknologi Informasi yang sangat pesat telah merambah setiap aspek kehidupan manusia. Tak terkecuali untuk kepentingan industri dan perdagangan, dimana perusahaan-perusahaan membutuhkan pertukaran data dengan supplier, konsumen maupun perusahaan lainnya. Pertukaran data tidak lagi dilakukan dengan media cetak tetapi dengan menggunakan data elektronik dengan format dokumen tertentu. Salah satu format dokumen elektronik adalah XML. XML merupakan sebuah bahasa markup yang diterima sebagai standar pertukaran data di dalam lingkungan internet.

Data-data yang akan dipertukarkan biasanya adalah data transaksi yang ada dalam basis data. Untuk itu maka diperlukan perangkat lunak yang mampu mengkonversikan data dari basis data ke dalam file XML dan sebaliknya.

Penelitian ini membangun perangkat lunak Sistem Pertukaran Data Antar Basis Data Dengan XML yang mampu mengkonversikan data dari berbagai macam basis data ke dalam file XML dan mampu mengkonversikan data dari file XML ke dalam berbagai macam basis data berbasis Windows, terutama Microsoft Acces, Microsoft SQL Server dan Oracle. Perangkat lunak ini juga dilengkapi dengan fungsi pengelolaan basis data yang akan mempermudah administrator basis data untuk mengelola basis data. Pembangunan perangkat lunak ini menggunakan pendekatan Object Oriented Software Development (OOSD).

Kata kunci: Pertukaran Data, XML, OOSD, EDI, Database

1. Pendahuluan

Keberadaan Teknologi Informasi tidak lagi dapat dipisahkan dari kehidupan manusia. Teknologi Informasi telah menjadi bagian dari kehidupan manusia dari persoalan rumah tangga, perdagangan sampai persoalan peluncuran wahana ilmu pengetahuan ke luar angkasa. Khusus untuk keperluan perdagangan maka Teknologi Informasi merupakan hal mutlak yang harus dipenuhi oleh semua perusahaan agar bisa bersaing dengan kompetitornya.

Dewasa ini, pertukaran data dalam sebuah perusahaan menjadi hal yang pokok yang harus dilakukan, sebagai contoh dalam penerapan *B2B Commerce (Business to Business Commerce)*. Pertukaran data menjadi hal yang penting bagi sebuah perusahaan bisnis karena mempermudah perusahaan untuk melakukan hubungan dengan supplier maupun dengan konsumen mereka. Dengan pertukaran data maka pencatatan transaksi dapat dilakukan secara elektronik secara penuh dan *online* yang memudahkan transaksi dan pengolahan datanya. Pertukaran data juga dapat berfungsi sebagai media untuk penyebaran informasi.

Salah satu format data pertukaran data secara elektronik adalah menggunakan format XML. Penggunaan XML di dalam metode pertukaran data secara elektronik mempunyai banyak keuntungan karena XML saat ini merupakan format standar dalam pertukaran data dan berorientasi pada penyimpanan data. Ada banyak kelebihan yang dimiliki XML sehingga banyak perusahaan yang

menggunakan XML sebagai dokumen untuk pertukaran data.

Pada umumnya, data-data suatu perusahaan tersimpan di dalam suatu basis data, sehingga untuk keperluan pertukaran data, perusahaan mengambil data dari basis data yang dimiliki baik itu katalog produk, transaksi, maupun lainnya. Jika perusahaan ingin memakai format dokumen XML di dalam pertukaran data, tentunya perusahaan harus mengkonversikan terlebih dahulu data-data di dalam basis data ke dalam format XML. Untuk data yang tidak terlalu banyak, hal tersebut tentunya tidak terlalu menjadi masalah, tetapi untuk data yang cukup banyak dan kompleks tentunya hal ini menjadi masalah besar untuk perusahaan. Agar lebih mempermudah suatu perusahaan dalam mengimplementasikan pertukaran data secara elektronik dengan format dokumen XML, perlu dibuat sebuah aplikasi yang dapat mengkonversikan data-data dari basis data ke dalam file XML.

Pada tulisan ini, akan dijelaskan konsep aplikasi yang dapat mengakomodasi kebutuhan tersebut. Hal ini akan sangat menguntungkan bagi perusahaan karena adanya otomatisasi dalam konversi data dari basis data ke dalam XML maupun sebaliknya. Aplikasi konversi data dari basis data ke dalam file XML, maupun file XML ke dalam basis data ini dibuat sebagai jembatan terutama bagi perusahaan yang menerapkan metode pertukaran data elektronik. Aplikasi yang dibuat juga dilengkapi dengan fasilitas untuk pengelolaan data di dalam suatu basis data, terutama data yang akan dikonversikan ke dalam file XML.

2. Extensible Markup Language (XML)

Extensible Markup Language (XML) adalah bahasa yang berbasis teks yang memungkinkan kita untuk menyimpan data di dalam format yang terstruktur. XML merupakan subset dari SGML (*Standard Generalized Markup Language*). SGML adalah standar *markup language* berdasarkan ISO 8879.

Dokumen yang dibuat dalam standar SGML memiliki definisi struktur dokumen yang dinyatakan dalam DTD (*Document Type Definition*)[1]. Struktur dokumen ditandai dengan *markup* berupa *tag*. *Tag* digunakan untuk membatasi elemen dokumen, yang penamaannya dapat ditentukan sendiri sehingga dapat digunakan untuk memberi semantik data. Dengan keleluasaan penggunaan tag ini, XML memungkinkan kita untuk menyimpan data secara terstruktur dan mudah dimengerti.

Sebuah dokumen XML terdiri dari 2 bagian utama, yaitu prolog dan elemen dokumen (juga dikenal dengan elemen root). Prolog berisi deklarasi tipe dokumen dan satu atau lebih instruksi pemrosesan. Elemen dokumen adalah sebuah elemen tunggal yang disebut dengan elemen root, yang dapat berisi elemen-elemen tambahan. Berikut ini contoh sebuah dokumen XML

```
<?xml version="1.0"?>
<!--File Name : Inventory.XML-->
<Inventory>
  <Book>
    <Title>The adventures of
Huckleberry Finn</Title>
    <Author age = "58">Mark
Twain</Author>
    <Price>$8.49</Price>
    <Pages>290</Pages>
  </Book>
</Inventory>
```

Gambar 1. Contoh Dokumen XML

Pada contoh tersebut, 2 baris pertama termasuk prolog kemudian *inventory* adalah elemen dokumen atau elemen root. *Book*, *Title*, *Author*, *Price* dan *Pages* adalah nama elemen tambahan dalam elemen root. Untuk elemen *author* terdapat atribut *age*. Elemen dalam dokumen XML berisi informasi dokumen dan mereka menandakan struktur logika informasi. Elemen tersebut disusun dalam struktur hirarki dalam bentuk pohon, dengan elemen yang bercabang dalam elemen yang lain. Semua elemen yang ada harus tersarang terhadap elemen dokumen atau elemen root.

Di dalam XML dimungkinkan juga penggunaan DTD (*Document Type Definition*) untuk menjaga validitas dari dokumen XML yang dibuat. Mengingat data yang akan kita konversikan berasal dari basis data, penggunaan DTD di dalam dokumen XML juga akan sangat menguntungkan karena kita dapat menjaga konsistensi dan integritas

data, baik dari struktur maupun tipe data yang digunakan.

Secara umum, dokumen yang seimbang jumlah *start-tag* dan *end-tag* nya disebut dengan *well-formed document*. Berdasarkan spesifikasi XML, *well-formed document* harus memenuhi kriteria sebagai berikut [4]:

- Dokumen dapat diturunkan dari aturan produksi dengan label *document* pada spesifikasi. Hal ini berarti dokumen dapat memiliki satu atau lebih elemen. Di antara elemen-elemen tersebut terdapat, ada tepat satu elemen yang disebut *root* atau *document element*. Untuk seluruh elemen, jika *start-tag* nya ada di dalam sebuah elemen, maka *end-tag* nya harus berada pada elemen yang sama.
- Dokumen memenuhi seluruh batasan *well-formedness* yang dinyatakan pada spesifikasi.
- Setiap *parsed-entities* yang diacu secara langsung maupun yang tidak langsung harus merupakan *well-formed document*.

Berikut adalah beberapa keuntungan penggunaan XML:

- Dokumen XML dapat digunakan di lingkungan internet dengan mudah tanpa memerlukan banyak perubahan konfigurasi pada aplikasi internet yang ada.
- Parser XML dapat diperoleh dengan mudah karena spesifikasi XML dibuat dalam bahasa formal yang mudah diterjemahkan ke dalam bahasa pemrograman.
- Dokumen XML dapat dimengerti dengan mudah dan jelas karena berupa arsip teks terstruktur, dengan elemen yang dibatasi tag yang didefinisikan sedemikian rupa sehingga mencerminkan semantik data.
- Spesifikasi XML telah didefinisikan oleh W3C (*World Wide Web Consortium*) dan dapat dimanfaatkan sebagai acuan pembuatan dokumen dan pemroses dokumen XML.
- XML adalah format yang tidak tergantung pada platform tertentu, sehingga akan menjadi keuntungan yang besar bagi perusahaan-perusahaan yang mempunyai platform sistem komputer yang berbeda.

3. Analisis dan Perancangan

Perangkat lunak ini diberi nama Sistem Pertukaran Data Antar Basis Data Dengan XML (SIPEDAFIX). Perangkat Lunak ini dibangun dengan menggunakan paradigma Object Oriented Software Engineering dengan menggunakan notasi UML.

3.1 Analisis

Fasa analisis dilakukan dengan menganalisa kebutuhan fungsional yang akan diimplementasikan

dan dilanjutkan dengan pembentukan kelas yang akan digunakan.

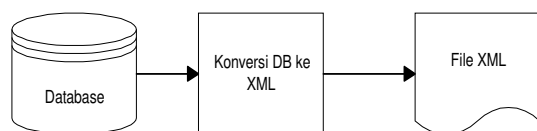
3.2 Kebutuhan Fungsional

Perangkat lunak ini adalah perangkat lunak yang dapat digunakan untuk memigrasikan data dari basis data tertentu ke dalam basis data yang lain dengan perantaraan file XML. Perangkat lunak ini dapat melakukan koneksi dengan basis data dalam platform windows, terutama basis data yang dapat terkoneksi dengan Microsoft Datalinks atau ODBC, seperti Microsoft SQL Server, Oracle, Microsoft Access, dan lain-lain.

Perangkat Lunak ini akan mengkonversikan data dari basis data yang dipilih ke dalam file XML maupun sebaliknya. Data yang dikonversi adalah data yang ada pada kolom pada setiap tabel yang ada di basis data yang dipilih oleh pengguna. Dalam hal ini pengguna dapat menentukan tabel mana saja atau kolom mana saja dari basis data yang akan dikonversikan.

Fungsi-fungsi yang dibutuhkan di dalam perangkat lunak ini antara lain:

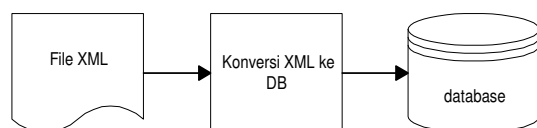
- a. Konversi data dari basis data tertentu ke dalam file XML



Gambar 2. Konversi data dari basis data tertentu ke dalam file XML

Fungsi ini digunakan untuk mengkonversi data pada basis data ke dalam file dengan format XML. Langkah penulisan XML dapat dengan menggunakan metode penulisan file biasa namun dengan format XML dan tersimpan ke dalam file XML, atau dengan adanya perkembangan lebih lanjut, misalnya ADO.NET dapat kita gunakan fungsi pembentukan XML yang telah dimilikinya.

- b. Konversi data dari file XML ke dalam basis data tertentu

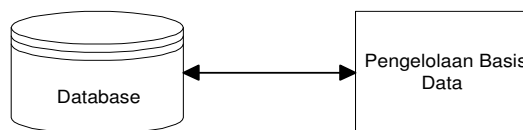


Gambar 3. Konversi data dari file XML ke dalam basis data tertentu

Fungsi kedua ini kita butuhkan untuk dapat membaca file yang berformat XML dan kemudian terkoneksi ke basis data untuk menuliskan data yang telah dibaca dari file XML ke dalam basis data. Format XML memungkinkan kita untuk melakukan

pembacaan dengan parser, sehingga akan lebih memudahkan kita dalam melakukan pencarian data sesuai dengan tag-tag yang ada di dalam format XML.

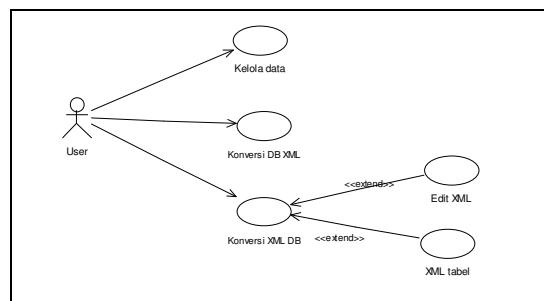
- c. Pengelolaan data yang ada di dalam basis data dengan Sql Builder.



Gambar 4. Pengelolaan data yang ada di dalam basis data dengan Sql Builder

Fungsi ketiga ini adalah fungsi tambahan jika kita menginginkan untuk terkoneksi ke basis data dan dapat melakukan manipulasi maupun pengelolaan terhadap basis data. Pada dasarnya fungsi ini cukup mudah dibuat jika kita telah dapat membuat fungsi pertama dan kedua.

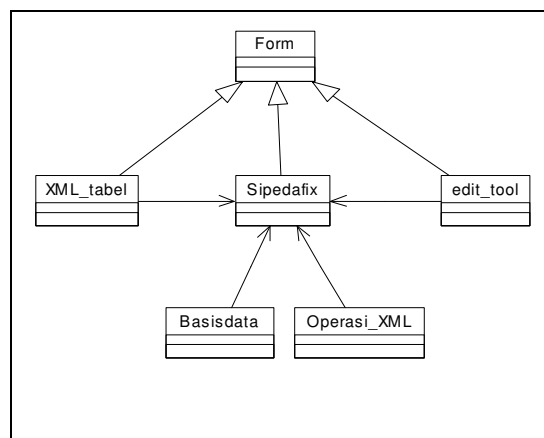
Dari kebutuhan fungsi tersebut, maka dapat dibentuk use case diagram sebagai gambaran akan fungsi yang harus dimiliki dan interaksinya dengan user



Gambar 5. Use Case Diagram SIPEDAFIX

3.3 Perancangan

Class Diagram SIPEDAFIX



Gambar 6. Class Diagram SIPEDAFIX

Penjelasan mengenai kelas-kelasnya adalah sebagai berikut:

1. Kelas SIPEDAFIX
Kelas SIPEDAFIX merupakan kelas pembentuk Graphical User Interface sebagai jembatan antara user dengan sistem. Kelas ini menjadi interface utama dari perangkat lunak SIPEDAFIX dan merupakan turunan dari kelas System.Windows.Forms.
2. Kelas XML_tabel
Kelas XML_tabel merupakan kelas interface untuk menampilkan file XML ke dalam bentuk table-tabel sehingga user dapat mempunyai sedikit gambaran dari hasil konversi XML ke DB yang akan dihasilkan. Kelas ini juga merupakan turunan dari kelas System.Windows.Forms
3. Kelas Edit_tool
Kelas edit_tool merupakan kelas interface untuk menampilkan file XML ke dalam textbox sehingga user dapat mengedit file XML tersebut. Ada beberapa fungsi ataupun operasi dasar pengeditan teks yang dimiliki oleh kelas ini antara lain cut, copy, edit, save dan print. Kelas ini akan dipanggil melalui kelas SIPEDAFIX.
4. Kelas Basisdata
Kelas Basisdata merupakan kelas yang digunakan untuk operasi terhadap basis data, baik itu pembukaan koneksi, proses SQL, maupun pembacaan dan manipulasi data yang ada di dalam basis data. Kelas ini akan dipanggil melalui kelas SIPEDAFIX.
5. Kelas Operasi_XML
Kelas Operasi_XML merupakan kelas yang digunakan untuk operasi terhadap file XML, baik itu pembacaan, maupun penulisan file XML. Kelas ini dipanggil melalui kelas SIPEDAFIX.

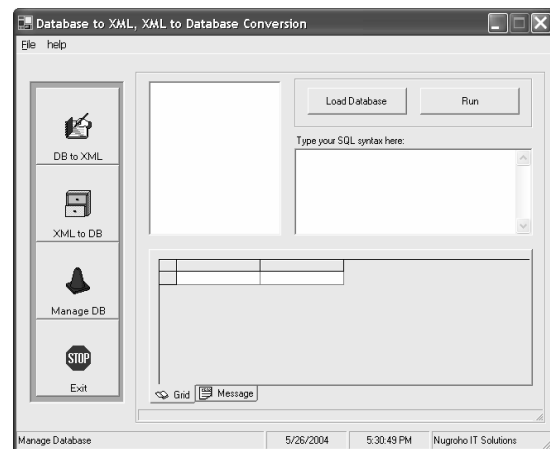
4. Implementasi

Sistem Pertukaran Data Antar Basis Data Dengan XML merupakan perangkat lunak yang berbasis *Graphical User Interface* (GUI). Perangkat lunak yang berbasis GUI mempermudah user untuk berinteraksi dengan perangkat lunak. Dalam implementasinya, beberapa kelas dari perangkat lunak ini merupakan kelas pembentuk tampilan GUI. Kelas-kelas pembentuk GUI pada perangkat lunak ini adalah kelas SIPEDAFIX, kelas XML_tabel dan kelas Edit_tool yang merupakan kelas turunan dari kelas System.Windows.Forms. Sedangkan kelas-kelas yang lain mempunyai fungsi-fungsi yang akan dipanggil melalui kelas pembentuk GUI.

Kelas terpenting dari aplikasi ini adalah kelas basisdata dan operasi_XML. Fungsi-fungsi utama yang dibutuhkan untuk membuat aplikasi konversi data dari file XML ke basis data maupun sebaliknya ada di kedua kelas ini. Jadi, sebenarnya aplikasi ini

dapat diintegrasikan ke dalam program lain yang membutuhkan fungsi-fungsi konversi data dari XML ke basis data maupun sebaliknya. Hal yang penting dari kedua kelas di atas adalah koneksi ke mesin basis data, manipulasi data di basis data, pembacaan dan penulisan file XML. Jika kita sudah terbiasa membuat aplikasi yang berkaitan dengan basis data, kesulitan yang dihadapi tidak terlalu besar karena untuk koneksi dan manipulasi data di dalam basis data konsepnya sama seperti jika kita membuat aplikasi basis data. Kita hanya perlu mempelajari format dokumen XML dan penulisan serta pembacaan file XML.

Berikut adalah *captured screen* dari perangkat lunak SIPEDAFIX.



Gambar 7. Tampilan Utama SIPEDAFIX

5. Penutup

Konsep aplikasi SIPEDAFIX ini merupakan pengembangan perangkat lunak yang mempunyai fasilitas untuk memigrasikan data dari suatu basis data ke dalam basis data yang lain dengan perantara format dokumen XML. Migrasi data antar basis data dengan perantara file XML merupakan solusi yang tepat karena format dokumen XML didukung oleh berbagai macam sistem dan platform, sehingga akan sangat membantu perusahaan-perusahaan yang menerapkan metode pertukaran data secara elektronik yang mempunyai sistem basis data yang berbeda.

Konsep aplikasi pertukaran data antar basis data dengan XML ini masih dapat dikembangkan lebih jauh sesuai dengan kebutuhan kita, terutama dalam kemudahan interaksi dan integrasi dengan aplikasi basis data lainnya. Perangkat lunak ini juga dapat dikembangkan ke dalam aplikasi berbasis web atau ke dalam aplikasi yang bersifat multi platform.

Daftar Pustaka

- [1] Bahrami, Ali., *Object Oriented System, Development*, Mc Graw Hill, Singapore, 1999.
- [2] Bray, Tim., et.al. *Extensible Markup Language (XML) 1.0 (Second Edition)*., <http://www.w3.org/TR/REC-XML>, 2000.
- [3] Bourret, Ronald., *XML and Databases*. <http://www.rpbourret.cpm/XML>, 2003.
- [4] Carter, Robert, *Elementary XML*, Microsoft Corporation, <http://www.microsoft.com/library/en-us/dnxml/html>, 1997.
- [5] NIIT, Courseware SN3., *Future Curriculum*, 2002.
- [6] Obasanjo, Dare., *Understanding XML*, Microsoft Corporation, <http://www.microsoft.com/library/en-us/dnxml/html>, 2003.
- [7] Pressman, Roger S., *Software Engineering : A practioner's Approach*, Mc Graw Hill, New York, 1998.
- [8] P.Sage, Andrew., *Systems Manajement for Information Technology and Software Enginnering*, John Wiley & Sons, Inc, 1995.
- [9] S. Davis, William., *Sistem Pengolahan Informasi Edisi II*, PT. Erlangga, 1991.
- [10] Whitten, Jeffrey L., Bentley, Lonnie D., Dittman, Kevin C., *Systems Analysis And Design Methods 5ed*, McGraw-Hill, 2001.
- [11] W. Jones, Gregory., *Software Engineering*, John Wiley & Sons, Inc, 1990.
- [12] Young, Michael J. *Step By Step XML*, Gramedia, Jakarta, 2001.