

Activex Data Objects dalam Pemrograman Delphi

Diaraya*

Abstrak

Activex Data Objects (ADO) adalah satu perangkat *Component Object Model*, yaitu objek untuk mengakses sumber data. ADO memungkinkan pengembang untuk menulis program yang mengakses data tanpa mengetahui bagaimana basis data diimplementasikan. Pada tulisan ini, akan dijelaskan teori beserta contoh program bagaimana cara mengakses basisdata dalam Delphi menggunakan komponen ADO.

Kata Kunci: ADO, basis data, Delphi.

1. Pendahuluan

Activex Data Objects (ADO) adalah antar muka berbasis objek untuk OLE DB yang bekerja dengan bahasa pemrograman. ADO dimasukkan sebagai bagian dari Microsoft Data Access Components (MDAC), bersama dengan OLE DB dan ODBC. Aplikasi ADO memungkinkan untuk menggunakannya dengan bahasa pemrograman yang mendukung COM (Teddy, 2002), seperti bahasa pemrograman Delphi. Model pemrograman ADO dibangun meliputi beberapa objek ADO yang digunakan untuk mengakses bermacam-macam sumber data.

Objek-objek ini menyediakan kemampuan untuk menghubungkan ke sumber data, query dan memperbaharui kumpulan record, dan melaporkan kesalahan. Delphi dapat menggunakan *Common Object Model*, dan salah satu kecanggihannya yang ditawarkan oleh Delphi adalah *Multi Tier Distributed Application Services* yang memungkinkan sebuah aplikasi untuk berkomunikasi dan berbagi data secara mendunia melalui jaringan internet (Jayanto, 1999).

Objek koneksi ADO digunakan untuk membuat koneksi terbuka pada sumber data, melalui koneksi ini digunakan untuk mengakses dan memanipulasi basisdata (ADO Tutorial, 2000). ADO adalah antar muka berbasis objek untuk OLE DB yang bekerja dengan bahasa yang mendukung *Component Object Model* (Gajic, 2000). Dengan menggunakan komponen ADO yang terdapat pada Delphi, dapat dibuat aplikasi basisdata yang tidak bergantung pada Borland Database Engine (Calvert, 1999).

2. Ruang Lingkup *Activex Data Objects* (ADO)

ADO terdiri dari tiga koleksi dan tiga objek. Koleksi ADO tersebut terdiri dari (Wikipedia, 1996):

1. Field

Koleksi ini berisi sekumpulan field objek yang dapat digunakan baik dalam objek recordset atau dalam sebuah objek record. Dalam sebuah object recordset, masing-masing field objek yang membentuk koleksi field sesuai dengan kolom di objek recordset. Dalam objek record, field bisa menjadi URL absolut atau relatif yang dapat digunakan untuk penyedia data semi-terstruktur seperti Microsoft OLE DB atau sebagai referensi yang terkait dengan objek rekam.

* Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin, Jl. Perintis Kemerdekaan Km.10, Tamalanrea Makassar.

Diaraya

2. Properties
Sebuah objek dapat memiliki lebih dari satu objek properti, yang terkandung dalam koleksi Properties objek.
3. Parameter
Sebuah *object command* dapat memiliki beberapa instruksi parameter untuk mengubah perilakunya yang telah ditetapkan, dan masing-masing objek parameter tersebut terkandung dalam objek koleksi parameter.

Sedangkan objek ADO tersebut terdiri dari:

1. Konektifitas
Objek konektifitas dalam lingkup obyek ADO berfungsi untuk menyimpan data melalui OLE DB. Objek tersebut menyimpan informasi tentang sesi dan menyediakan metode untuk menghubungkan ke penyimpanan data. Seperti beberapa penyimpanan data yang memiliki metode yang berbeda saat membuat koneksi, beberapa metode mungkin tidak didukung pada konektifitas untuk objek tertentu pada penyedia OLE DB.
2. Instruksi
Setelah menetapkan objek sambungan sesi ke sumber data, instruksi yang dikirim ke penyedia data melalui objek instruksi. Objek instruksi dapat mengirim query SQL langsung ke operator melalui penggunaan properti CommandText, mengirim permintaan parameter atau prosedur yang tersimpan melalui penggunaan objek parameter atau koleksi untuk menjalankan query dan mengembalikan hasil ke objek dataset melalui running query. Ada beberapa metode lain yang dapat digunakan di instruksi objek yang berkaitan dengan recordset.
3. Recordset
Recordset adalah sekelompok catatan, baik yang berasal dari tabel dasar atau sebagai hasil dari tabel query. Objek recordset berisi koleksi field dan koleksi properties. Koleksi field adalah seperangkat field, yang merupakan kolom yang sesuai dalam tabel, sedangkan koleksi properties adalah seperangkat properti, yang mendefinisikan fungsi tertentu dari penyedia OLE DB. Recordset memiliki banyak metode dan properti untuk memeriksa data yang ada didalamnya. Rekaman dalam recordset dapat diperbaharui dengan mengubah nilai-nilai dalam catatan dan kemudian memperbaharui rekaman tersebut, yang dinamakan dengan metode update batch. Adapun metode untuk menambahkan catatan baru dilakukan melalui fungsi add new dan kemudian melakukan perubahan atau melakukan metode update batch. Akan tetapi, jika untuk beberapa alasan penghapusan tidak dapat terjadi, seperti karena pelanggaran dalam integritas referensial, maka recordset akan tetap dalam mode edit setelah panggilan ke metode update. Dalam hal ini, programmer harus secara eksplisit memanggil fungsi untuk membatalkan pembaruan. Recordset juga dapat diperbarui dalam salah satu dari tiga cara, yaitu melalui pembaruan dengan immediate, melalui update batch, atau melalui penggunaan transaksi.

3. Basisdata Activex Data Objects (ADO)

3.1 Pembuatan Basisdata ADO

Sebelum menggunakan Activex Data Object, buatlah terlebih dahulu output yang diinginkan. Dari bentuk output rancangan tersebut, kemudian dibuat tabel databasenya. Berikut tahapan pembuatannya, yaitu :

1. Setelah tabel database dibuat, selanjutnya kita bekerja di program Delphi dengan membuat koneksi pada komponen Openpicturedialog, dengan cara klik dobel button Foto, isi dengan program:

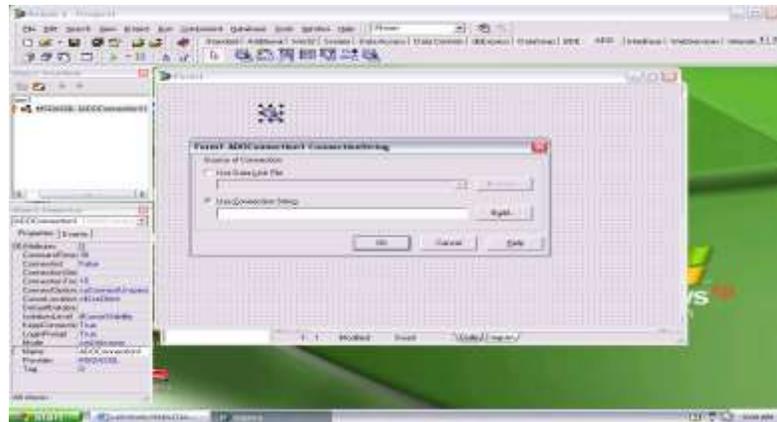

```
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
  If OpenPictureDialog1.Execute Then
    dbimage1.picture.loadfromfile(openpicturedialog1.filename);
```

Diaraya

- end;
2. Klik doble button Keluar, kemudian isi dengan program:

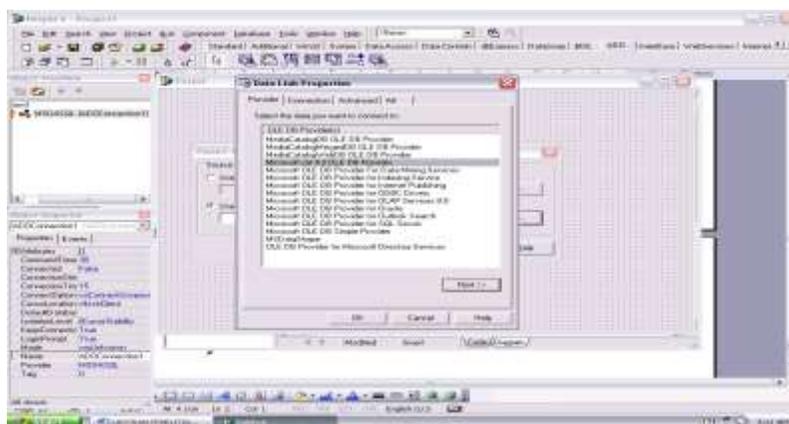

```
procedure TForm1.BitBtn2Click(Sender: TObject);
begin
  Application.Terminate;
end;
```
 3. Klik doble DBNavigator1, juga isi dengan program:


```
procedure TForm1.DBNavigator1Click(Sender: TObject; Button: TNavigateBtn);
begin
  Dbedit1.SetFocus;
end;
```
 4. Selanjutnya klik doble komponen ADO Connection, sehingga tampak seperti dalam Gambar 1.



Gambar 1. Komponen ADO Connection.

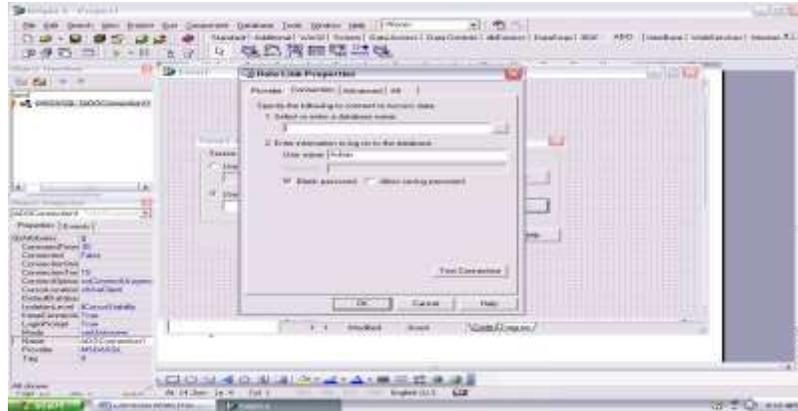
5. Klik tombol Build, sehingga tampak seperti dalam Gambar 2.



Gambar 2. Data Link Properties.

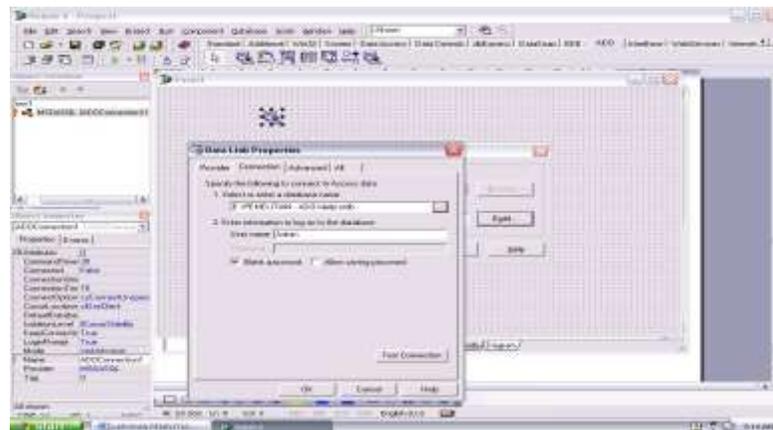
6. Selanjutnya klik tombol Next, sehingga tampak seperti dalam Gambar 3.

Diaraya



Gambar 3. Data Link Properties Lanjutan.

7. Insert tabel database yang telah dibuat dengan MS-Access, dengan mengklik double sehingga akan tampak seperti dalam Gambar 4.



Gambar 4. Link Database MS-Access.

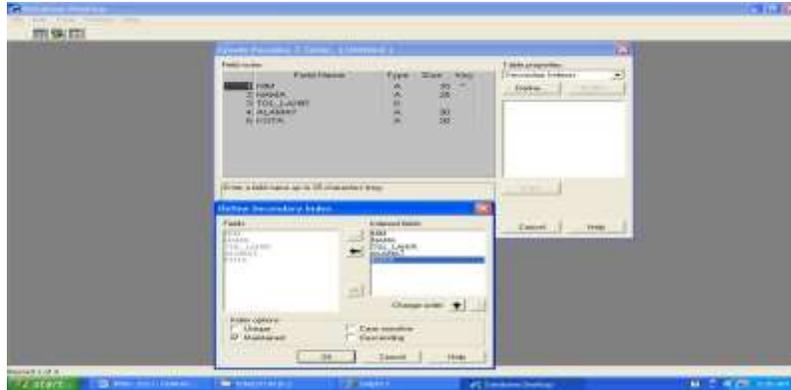
8. Klik Ok , sehingga terkoneksi database yang diinginkan menggunakan ADO.

3.2. Input Data Tabel dengan Delphi

Tahapan pemasukan data tabel adalah sebagai berikut:

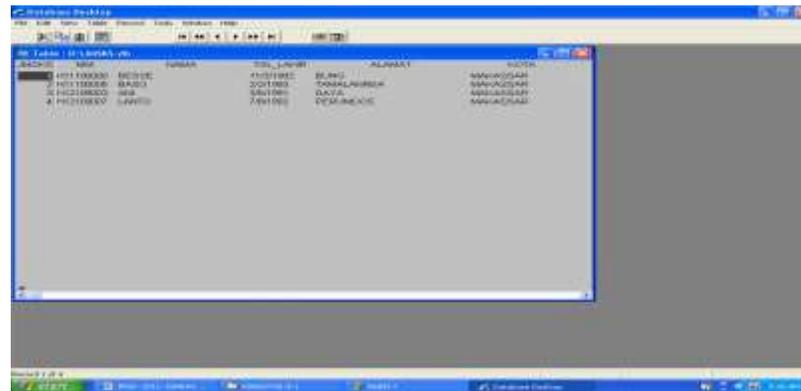
1. Klik Menu Tools Delphi
2. Pilih Database Desktop
3. Klik Menu File
4. Pilih New
5. Pilih Tabel → kemudian OK
 - Klik table properties
 - Pilih Secondary Indexes
 - Klik Define
 - Klik OK
6. Lengkapi Tabel sehingga seperti dalam Gambar 5.

Diaraya



Gambar 5. Editor Input Data.

7. Klik Save As
8. Pilih directory yang diinginkan, simpan file data
9. Klik menu File → Pilih Open → Klik Table
10. Load file data
11. Lengkapi tabel sehingga seperti dalam Gambar 6.



Gambar 6. Pembuatan Tabel Delphi.



Gambar 7. Output Program.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Penggunaan ADO dalam pemrograman Delphi dipusatkan pada komponen ADO Express pada tab ADO dan palette komponen. Salah satu kelebihan aplikasi yang dibangun dengan menggunakan komponen ADO adalah tidak mengharuskan menggunakan Borland Database Engine dan hampir semua basisdata lain memungkinkan untuk diakses dengan komponen ini.

4.2. Saran

Pemrograman Delphi dengan *Activex Data Objects* (ADO) sebaiknya diperluas dengan menggunakan komponen ADO Extension untuk data definition language dan security.

Daftar Pustaka

- ADO Tutorial, 2000. Website: <http://www.w3schools.com/ado/default.asp>. Copyrights 1999-2010 by Refsnes Data, All Rights Reserved, diakses pada Oktober 2010.
- Calvert, C., 1999. Accessing databases using ADO and Delphi. Website: <http://edn.embarcadero.com/article/10270>, diakses pada Oktober 2010.
- Jayanto, 1999. *Membuat Aplikasi Database dengan Delphi*. PT. Elex Media Komputindo, Jakarta.
- Teddy, M., Agus, P., dan Josef, W., 2002. *Pemrograman Delphi dengan ADO Express, Mengakses Basisdata MS. Access*. Penerbit Informatika, Bandung.
- Wikipedia, 1996. Website: http://en.wikipedia.org/wiki/ActiveX_Data_Objects#ADO_objects, diakses pada Oktober 2010.
- Gajic, Z., 2000. Using ADO in a Delphi application. Website: <http://www.delphi.about.com/od/database/a/adodelphi.htm>, diakses pada Oktober 2010.