

# IMPLEMENTASI DATABASE PADA APLIKASI MULTIMEDIA DENGAN MACROMEDIA DIRECTOR DAN DATAGRIP

Amir F. Sofyan

## Abstraksi

Adanya sifat permanen dalam hal konten pada format CD-ROM –sebagai media distribusi aplikasi multimedia— menyebabkan langkanya penggunaan database dalam aplikasi multimedia. Macromedia Director sebagai software authoring multimedia pada dasarnya tidak mendukung konektifitas dengan database. Datagrip merupakan salah satu Xtras (ekstensi) dari Director yang mampu mengkoneksikan Director dengan database.

*Kata kunci:* multimedia, database

## Pendahuluan

Selama ini Director dengan bahasa script Lingo hanya mendukung database dengan tipe linear lists atau property lists yang terbatas. Meskipun demikian Director dapat dikoneksikan dengan database menggunakan Xtras yang dibuat oleh pihak ketiga. Datagrip adalah salah satu Xtras yang menggunakan Microsoft Jet Engine untuk berkomunikasi dengan Microsoft Access database. Tulisan berikut menjelaskan implementasi Datagrip pada Director.

## Pembahasan

Sebagai contoh kasus implementasi Datagrip pada Director berupa aplikasi daftar kontak yang berguna untuk mengelola daftar kontak fotografer/ pelukis beserta contoh karyanya. Langkah pertama adalah menyiapkan database dengan Microsoft Acces yang berupa dua tabel yang berisi daftar kontak dan daftar nomor telpon. Daftar telepon dipisahkan menjadi tabel tersendiri karena masing-masing fotografer/ pelukis mungkin saja memiliki beberapa nomor telepon (Gambar 1 dan 2). Untuk data seperti nama dan alamat menggunakan tipe data text, sementara untuk gambar dan suara menggunakan tipe data OLE Object.

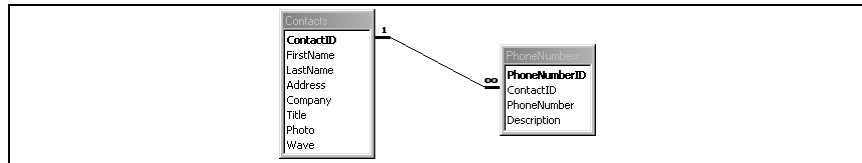
The image shows two table structures side-by-side. The first table is 'Contacts : Table' and the second is 'PhoneNumbers : Table'. Both tables have columns for 'Field Name' and 'Data Type'.

Field Name	Data Type
ContactID	AutoNumber
FirstName	Text
LastName	Text
Address	Text
Company	Text
Title	Text
Photo	OLE Object
Wave	OLE Object

Field Name	Data Type
PhoneNumberID	AutoNumber
ContactID	Number
PhoneNumber	Text
Description	Text

Gambar 1. Struktur tabel



Gambar 2. dan relasi antar tabel

Selanjutnya menyiapkan interface pada Director (Gambar 3) yang terdiri dari beberapa field untuk menampung database.

Gambar 2. Interface aplikasi database dengan Datagrip

Langkah berikutnya adalah membuka file database menggunakan fungsi DGOpendatabase pada Datagrip. Fungsi ini memerlukan satu parameter yaitu nama database yang akan dibuka.

```
set dbHandle = DGOpendatabase("Tutor.mdb")
```

Pada script di atas, file database terletak pada direktori yang sama dengan projector. Selain itu dapat juga dapat digunakan script untuk mengakses database dalam jaringan dengan menyebutkan nama komputernya.

```
set dbHandle = DGOpendatabase("\\FileServer\Tutor.mdb")
```

Langkah berikutnya, setelah membuka database, adalah membuat recordset untuk membaca data dari database. Recordset dapat berisi semua record dalam table atau beberapa record saja berdasarkan kriteria tertentu dengan SQL query seperti pada script berikut.

```
set rsHandle = DGCreateRecordset("SELECT * FROM Contacts",
dbHandle)
```

Atau dapat dilakukan dengan cara lain untuk menyeleksi nomor telepon dari ContactID tertentu, misalnya ContactID = "2".

```
set queryStr = "SELECT * From PhoneNumbers Where ContactID="
& CurContact
set rsHandle = DGCreateRecordset(queryStr, dbHandle)
```

PhoneNumberID	ContactID	PhoneNumber	Description
3	2	(990) 222-7400	Data
4	2	(800) 222-2222	Toll Free Number
5	2	(503) 432-2474	Home
6	2	(503) 209-3498	Fax
*	(AutoNumber)	0	

**Gambar 3.** Daftar nomor telepon untuk ContactID nomor 2

Setelah memiliki informasi yang dibutuhkan dari database maka langkah selanjutnya adalah menampilkannya pada stage. Pada layar utama terdapat delapan field yang akan menampung data dari database. Nama field pada stage berhungan dengan nama field pada database, seperti field untuk nama depan, nama belakang, alamat, perusahaan, jabatan dan contoh karya. Nomor telepon dan keterangan ditampung dalam list box.

Terdapat beberapa pendekatan untuk menampilkan informasi pada stage, yang salah satunya adalah dengan mengaitkan secara langsung antara cast member pada Director dengan field pada database menggunakan fungsi DGRSBindField. Pada fungsi DGRSBindField terdapat tiga parameter, yaitu pertama nama cast member, kedua, field pada database, dan ketiga handle recordset.

```
DGRSBindField("FirstName_Field", "FirstName", rsHandle)
DGRSBindField("LastName_Field", "LastName", rsHandle)
DGRSBindField("Company_Field", "Company", rsHandle)
DGRSBindField("Title_Field", "Title", rsHandle)
```

Pendekatan lainnya –untuk menampilkan informasi pada stage—adalah menggunakan DGRSGetFieldValue(). Hal ini dilakukan apabila data tersimpan pada beberapa field yang berbeda dan ingin ditampilkan bersamaan. Sehingga data harus diambil kemudian diubah menjadi string.

```
set txt = DGRSGetFieldValue("Address", rsHandle)
set the text of member "Address_Field" = txt
```

Kemudian langkah selanjutnya adalah menampilkan daftar nomor telepon pada list box. Akan tetapi Director hanya menyediakan list box sederhana yang terdiri dari satu kolom. Untuk mengatasi hal ini maka dapat digunakan fungsi Grip pada Datagrip yang mendukung multi-column list box. Pertama memanggil fungsi DGCreateGrid untuk membuat grid, dan kemudian memanggil fungsi DGGridAddField untuk tiap kolom yang akan ditampilkan. Selanjutnya fungsi DGGridUpdate untuk mengisi data pada grid.

```
set gridHandle = DGCreateGrid("PhoneNum_Field", tmpRS)
DGGridAddField("Description", 13, gridHandle)
DGGridAddField("PhoneNumber", 14, gridHandle)
DGGridUpdate(gridHandle)
```

Berikutnya untuk menambah record dapat digunakan beberapa fungsi. Misalnya fungsi DGRSAddNew untuk menambah record baru yang kosong, dan DGRSEdit untuk mengedit record yang ada.

```
on CreateNewRecord
  global rsHandle
  if DGRSGetEditMode(rsHandle)=2 then
    SaveCurrentData()
  end if
  DGRSAddNew(rsHandle)
end
```

Untuk membaca data dari field dapat digunakan fungsi DGRSGetFieldValue, dan untuk memasukkan data ke dalam field digunakan fungsi DGRSSetFieldValue. Selanjutnya untuk menyimpan record baru atau yang telah diedit ke dalam database maka digunakan fungsi DGRSUpdate.

```
on SaveDBField fieldName
  global rsHandle
  DGRSSetFieldValue(fieldName, the text of member(fieldName &
  "_Field"),rsHandle)
end
```

Sampai saat ini hanya daftar kontak pertama yang terlihat. Untuk itu perlu ditambahkan fungsi baru agar dapat melihat kontak lainnya, yaitu dengan nama GoToNextRecord dan GoToPrevRecord.

```

on GoToPrevRecord
  global rsHandle
  if DGRSGetEditMode(rsHandle) = 2 then
    SaveCurrentData()
  end if
  DGRSMovePrev(rsHandle)
  if DGRSIsBOF(rsHandle) then
    DGRSMoveNext(rsHandle)
  end if
end

```

Bagian berikutnya adalah menambah nomor telepon sebagai salah satu fitur penting pada aplikasi ini.

```

set descStr = the text of member "tmpDesc_Field"
set NumStr = the text of member "tmpNum_Field"
set id = DGRSGetFieldValue("ContactID", rsHandle)
if id <> "#ERROR#" then
  set tmpRS = DGCreateRecordset("SELECT * FROM PhoneNumbers Where
ContactID =" >>
  && id, dbHandle)
  DGRSAddNew(tmpRS)
  DGRSSetFieldValue("ContactID", id, tmpRS)
  DGRSSetFieldValue("Description", descStr, tmpRS)
  DGRSSetFieldValue("PhoneNumber", numStr, tmpRS)
  DGRSUpdate(tmpRS)
  DGRSClose(tmpRS)
end if

```

### Kesimpulan

Dengan adanya Xtras Datagrip maka aplikasi multimedia menggunakan Director dapat lebih fleksibel dalam pengelolaan kontennya. Datagrip mampu menampilkan data dengan tipe teks, gambar maupun suara. Apabila terjadi perubahan konten –seperti penambahan, pengurangan atau penyuntingan—maka cukup diubah pada file database, dan tidak perlu mengubah aplikasi multimedia atau tidak harus membangun dari awal lagi.

### Daftar Pustaka

*Datagrip User Manual*, Oregon, 1998

[www.datagrip.com](http://www.datagrip.com)