

【A5】 Delphiテクニカルセッション

「Delphi+Visual LiveBindingによる データベースアプリケーション開発」

株式会社ドリームハイブ
代表取締役 ITコンサルタント
山本 悟



自己紹介

- 名前: 山本 悟 (やまもと さとる)
 - 代表取締役 & ITコンサルタント
- 会社: 株式会社ドリームハイブ
 - 会社URL : <http://www.dreamhive.co.jp/>
 - フリーソフトなどを配信 : <http://dhive.jp/>
 - 山本のブログ : <http://dhive.jp/blog/yama/>
 - facebook : <http://www.facebook.com/kryu2>
- 山本はこんな人:
 - 17歳からIT業界へ
 - Delphi は1.0からの親友
 - ドリームハイブの経営、ITコンサルティング、システム開発、スピーカーなどが主な仕事

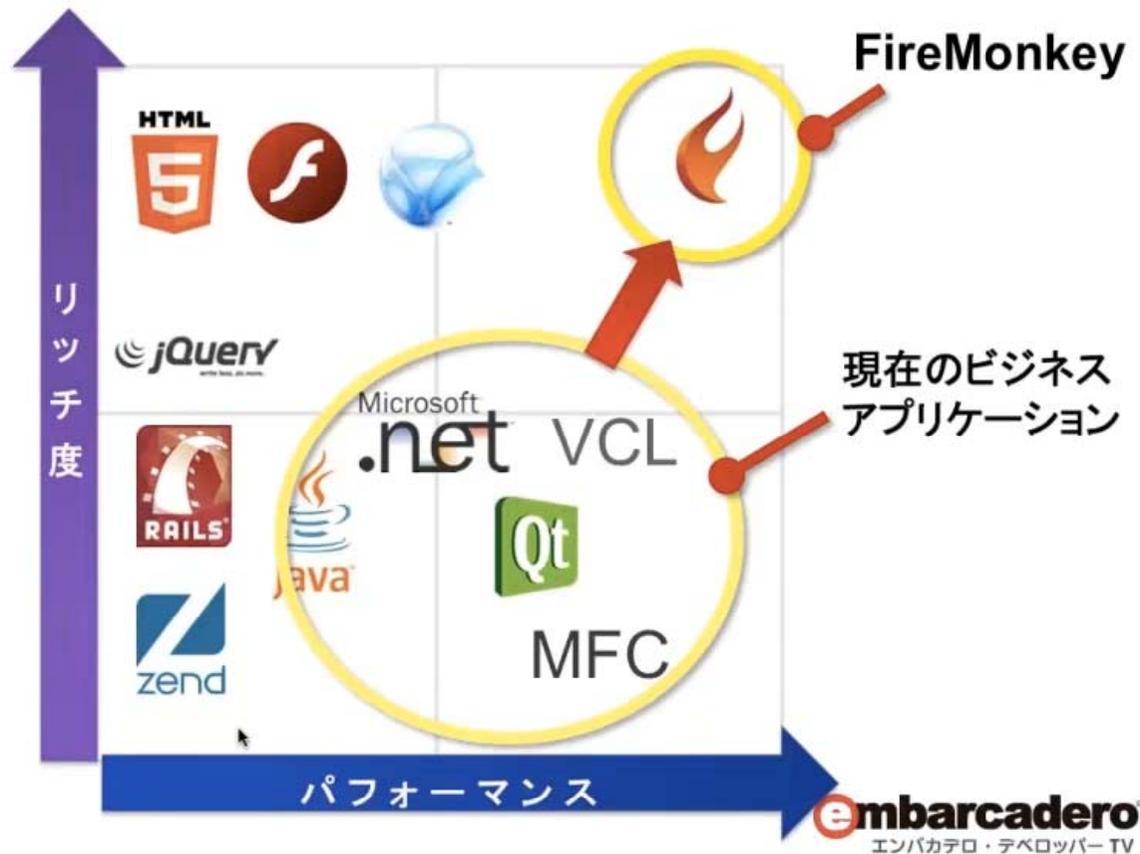
本日の流れ

- VCLとFMとVisual LiveBinding
- 接続の基本のデモ
- 双方向の接続のデモ
- フィールドを利用した接続のデモ
- DataSourceと接続のデモ
- 他の形式のデータと接続のデモ
- まとめ

- このセッションは、テクニカル・セッションです
- RAD StudioでLiveBindingを用いたアプリケーションの構築方法について、私の主観と経験に基づいてお話しいたします
- あなたにとって最適解では無いかもしれませんが、参考になると思います

VCLとFireMonkey

- RAD Studio XE2からフレームワークに VCLとFireMonkey(XE3では FM2)が追加されました

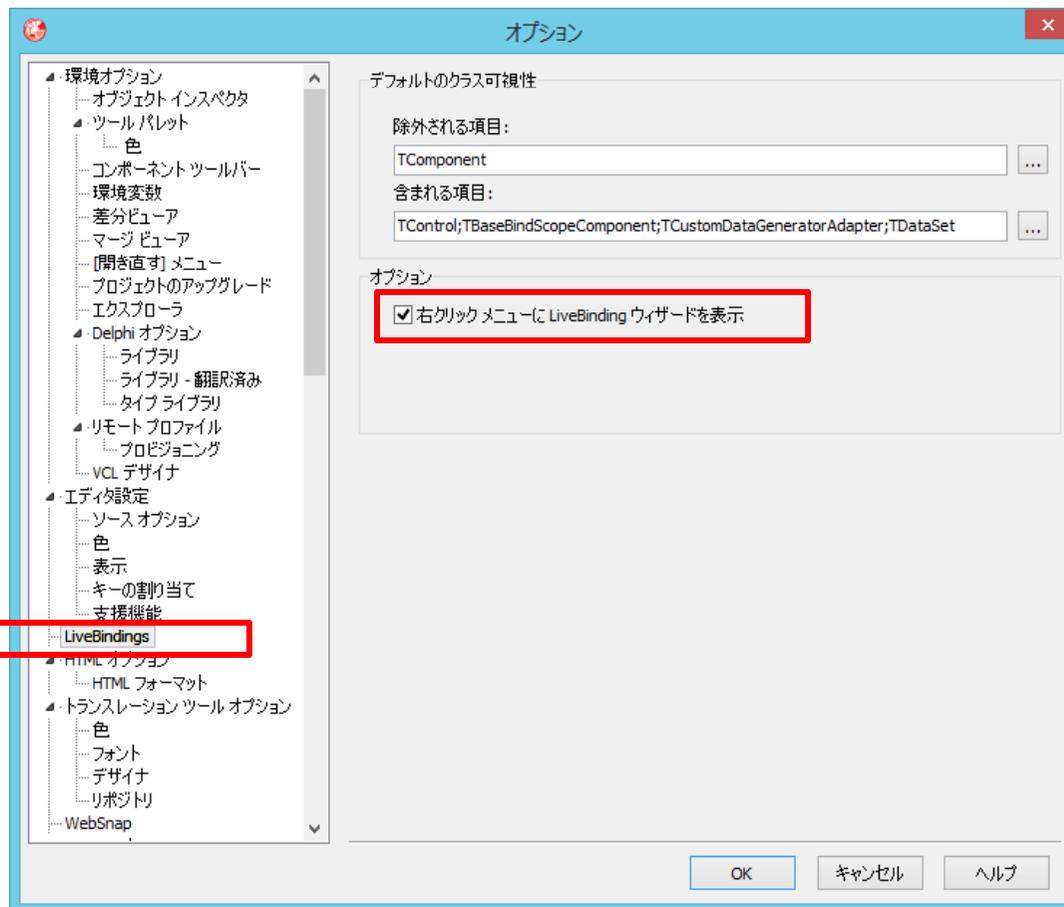


LiveBindingとは

- VCL/FMのどちらにもLiveBindingがありますが、FMはクロスプラットフォームです
- LiveBindingの概要
 - バインディング式と呼ばれる、
単方向にも双方向にもできる関係式に基づいている
 - LiveBinding は伝播する
 - コントロール オブジェクトとソースオブジェクト
- バインディングの作り方
 - LiveBindingデザイナーかLiveBindingウィザード
 - クイックバインディングのみ作成できる
 - LiveBindingコンポーネントの利用
 - バインディング式などを指定できる
 - 注意
 - 1つのコントロールを2つのコンポーネントプロパティにリンクすることはできない
 - コントロールと両方のコンポーネントプロパティを1つのフィールドにリンクすることは可能
- と言う事で、本日はLiveBindingについて話します

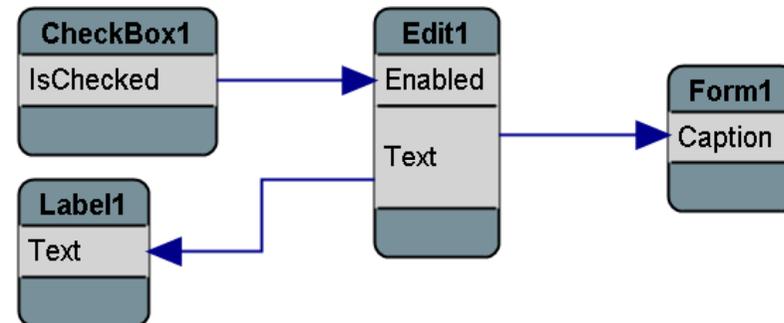
LiveBindingのための設定

- オプション設定
右クリックメニューにLiveBindingウィザードを表示



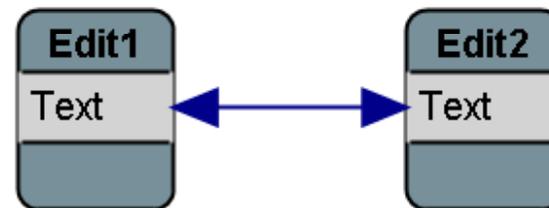
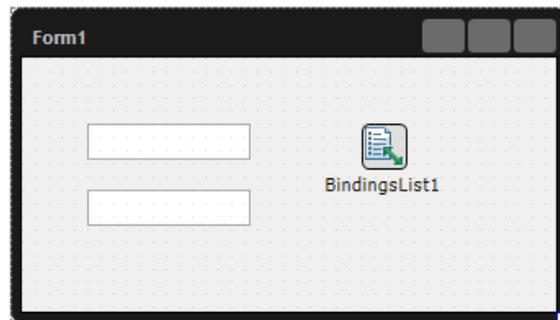
接続の基本

- LiveBindingはコンポーネントのプロパティを動的に結びつける
 - EditとLabelを結びつけるなどの基本のデモ01



双方向の接続

- 双方向に繋げることも
 - EditとEditを接続するデモ02
- キーワード
 - BindingsList1.Notify()
 - Direction := dirBidirectional

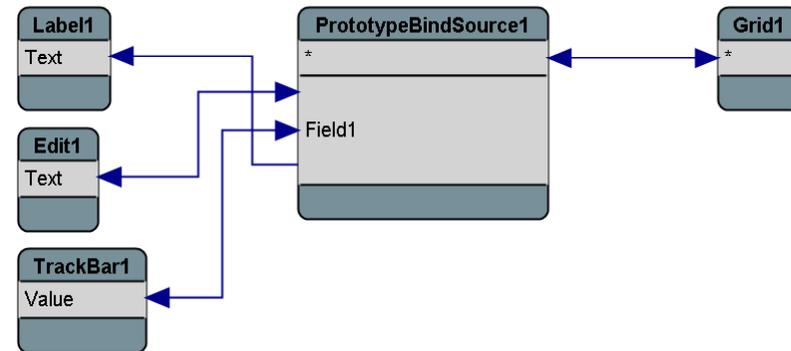
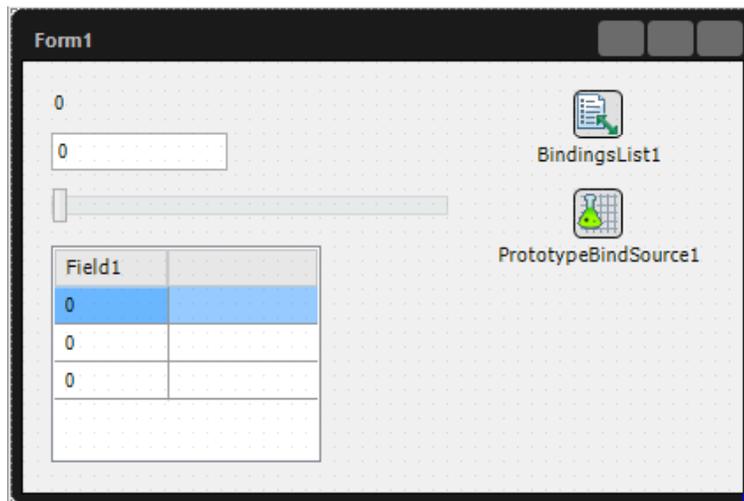


フィールドを利用した接続1

- PrototypeBindSourceとは
 - PrototypeBindSourceとは、アプリケーションのテストのために様々なフォーマットのサンプルデータを提供する機能をもつコンポーネントです。

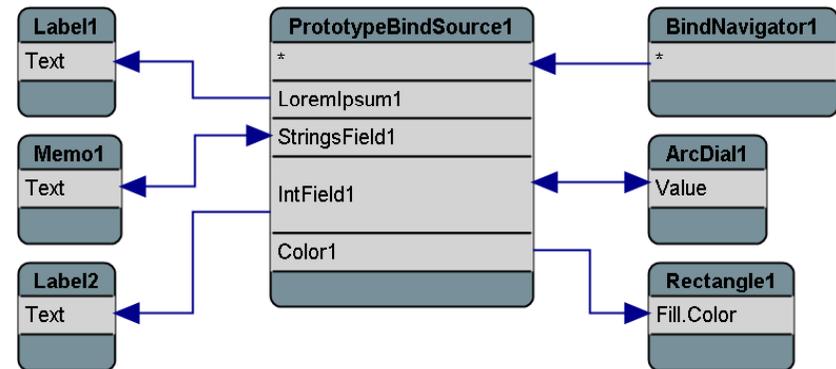
フィールドを利用した接続2

- PrototypeBindSourceを使い、フィールドを経由して様々なコンポーネントをリンクするデモ03



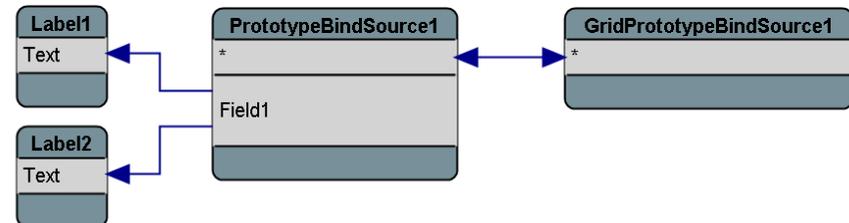
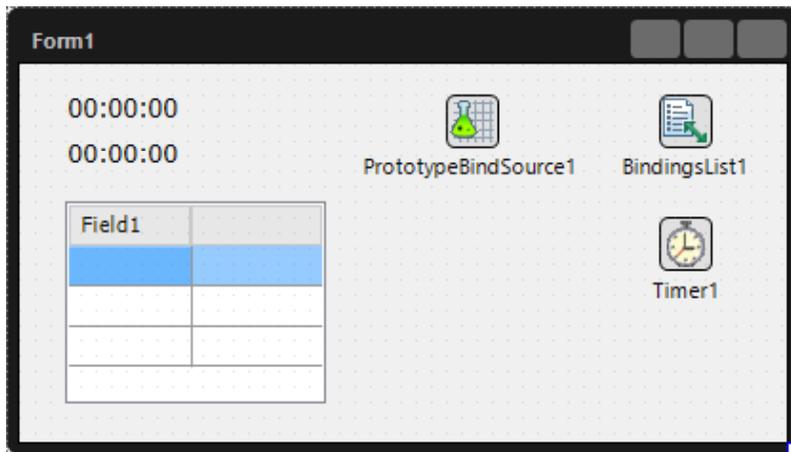
フィールドを利用した接続3

- Generatorを使ったサンプルデータを利用するデモ04



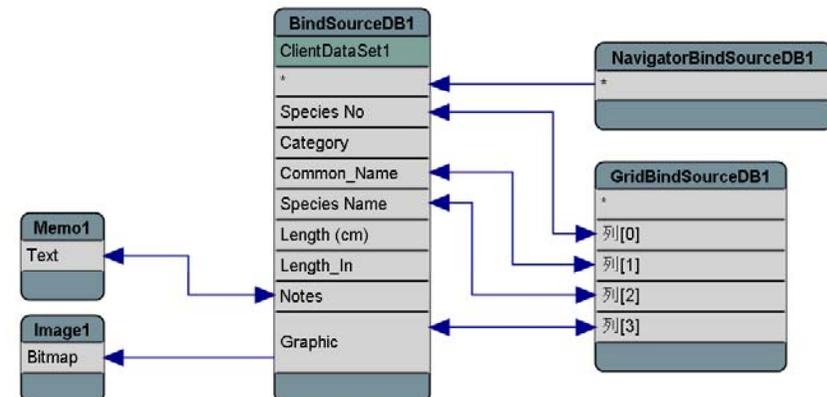
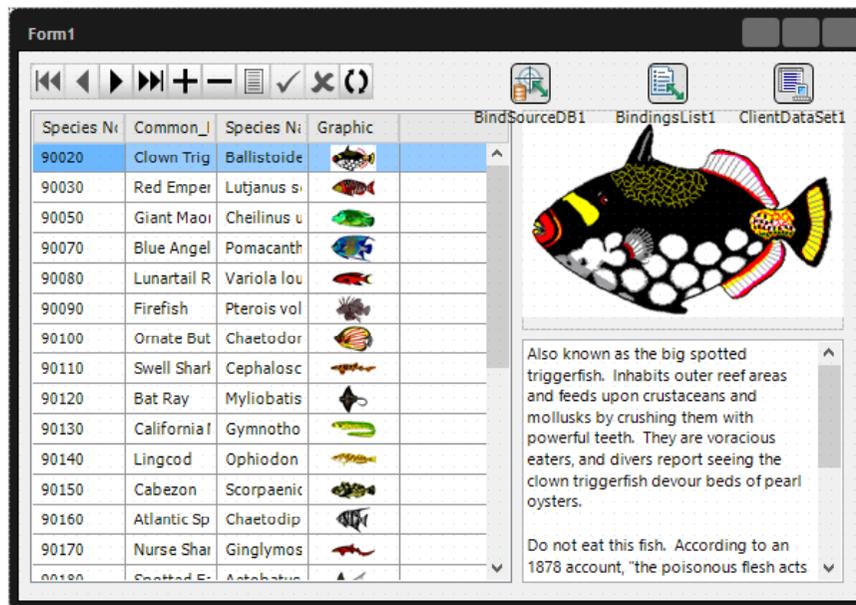
フィールドを利用した接続4

- PrototypeBindSourceのデータをコードで書き換えるデモ05



DataSourceと接続1

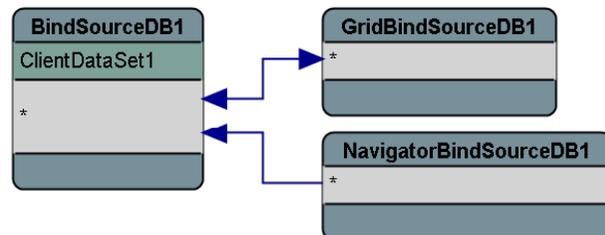
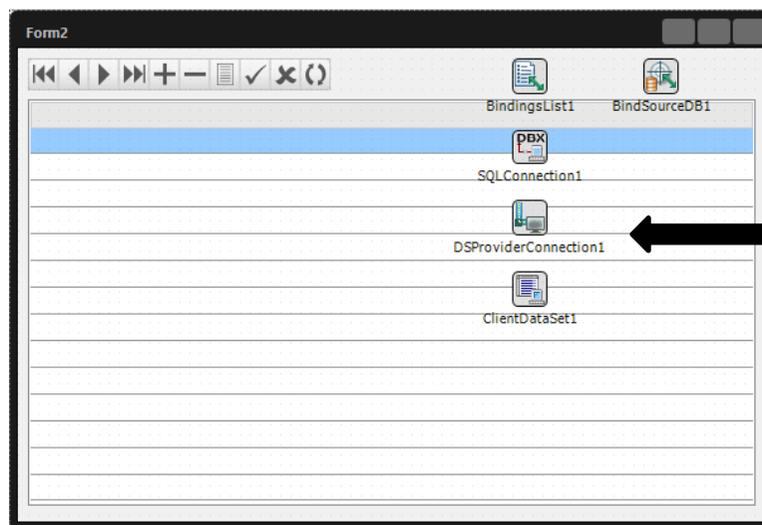
- LiveBindingはDataSourceにも繋がります
 - DataSourceとGridを接続するデモ06
- 注意
 - LinkGridToDataSourceのColumnsを弄るときはAutoActivateをFalseに



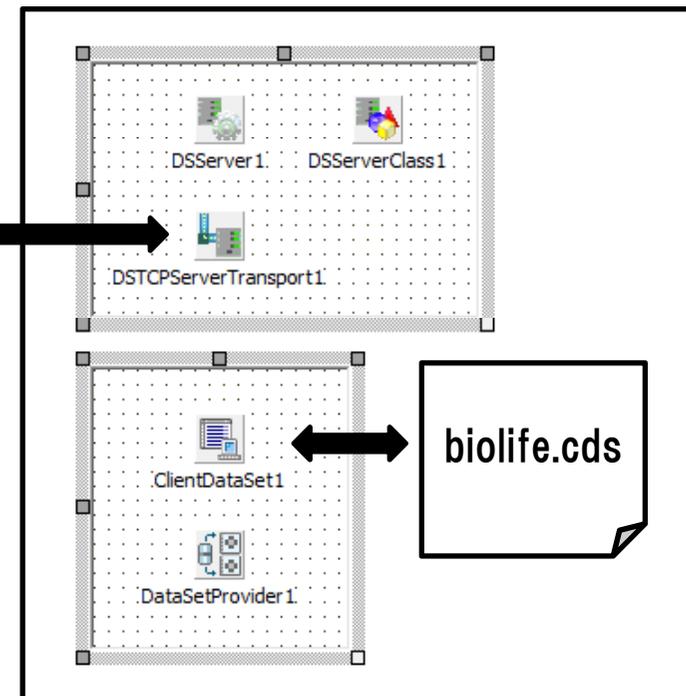
DataSourceと接続2

- DataSnapを利用すればネットワーク経由の接続もOK
 - DataSnap経由でGridを表示するデモ07

DataSnapクライアント

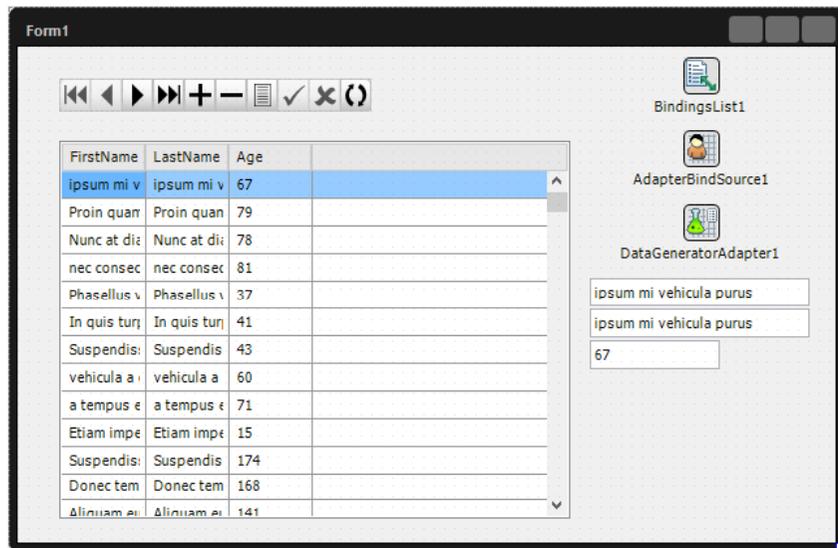


DataSnapサーバー

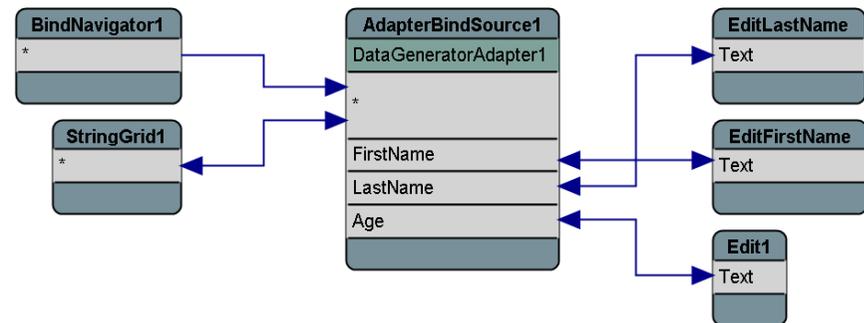


他の形式のデータと接続1

- アダプタを使えば自作クラスとやりとりもできます
 - 自作クラスを
TAdapterBindSource経由でリンクさせるデモ08



自作クラス: EmployeeAdaptee.pas



EmployeeAdaptee.pas

```
unit EmployeeAdaptee;

interface

type
  TEmployee = class(TObject)
  private
    FFirstName: String;
    FLastName: String;
    FAge: Byte;
  public
    constructor Create(const AFirstName, ALastName: String; const AAge: Byte); overload;
    property FirstName: String read FFirstName write FFirstName;
    property LastName: String read FLastName write FLastName;
    property Age: Byte read FAge write FAge;
  end;

implementation

constructor TEmployee.Create(const AFirstName, ALastName: String; const AAge: Byte);
begin
  inherited Create;

  FFirstName := AFirstName;
  FLastName := ALastName;
  FAge := AAge;
end;

end.
```

まとめ

- VCL + DBコンポーネントから
FireMonkey + Visual LiveBinding に移行することで
 - ✓ Windows / Mac のクロス環境で実行可能
 - ✓ 高品質なグラフィック / UIを利用可能
 - ✓ 表現力豊かなコンポーネントを使用可能
 - ✓ コンポーネントの選択肢が広がる
(DBコンポーネントだけに限定されない)
 - ✓ タブレットなどでも利用しやすいスタイルを選択可能
 - ✓ 将来的にモバイルなど追加のプラットフォームでも
利用可能(iOS / Androidが予定されている)
 - ✓ RDBMSだけでなく任意のデータ / オブジェクトを扱える



Q & A



終わりに

ご静聴いただきありがとうございました

メルマガもご登録ください(※期間限定)



<http://www.dreamhive.co.jp/24thDC/>

パスワード:24thDC