

MAXIMIZE THE  
BUSINESS VALUE  
OF SOFTWARE

# データベースアプリケーション構築技法

Delphi、C++Builderによるメンテナンス性を考慮した開発

第一章:アーキテクチャの理解

# Delphi / C++Builder アーキテクチャ

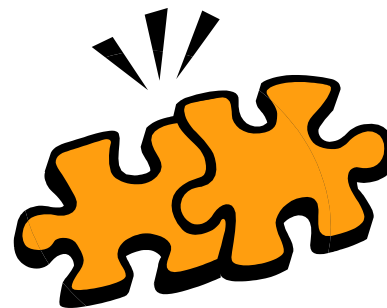
- ビジュアルに“RAD”を行なえるツール
  - RAD ≠ Visual Development
    - James Martin が1991年の著書で提案している開発手法
      - プロトタイピング / スパイラルアプローチ 等が特徴
  - 今の言葉でコンポーネント指向の開発を実現する仕組みがある
    - ここでのコンポーネント ≠ VCL コンポーネント
- Delphi / C++Builder は、これらを上手く利用するための仕組みを持つ
  - ビジュアル開発
    - 操作の結果、IDE が自動生成するコードが、再利用性を考慮したコードとなる
  - 豊富な機能を提供するクラス(コンポーネント)ライブラリ、VCL の存在
  - VCL 以外の標準的な共通部品 (ActiveX コントロール、DLL) も利用可能
    - また、これらの作成も可能

# VCL アーキテクチャ

- コンポーネント
  - 比較的、再利用性が高いプログラム部品
  - 一つもしくは複数のクラスからなる
- VCL コンポーネント
  - 基底クラス TComponent から派生させたクラス
    - VCL の約束事(フレームワーク)に準拠させていれば、Borland 純正、3rd Party 製品、自作等を区別なく同等に利用できる。
  - IDEのコンポーネントパレットに登録することで、ビジュアルな画面操作で再利用できる
  - ビジュアル開発可能だが、UI 作成だけではない
    - Web / Web Service / データベースアクセス、操作 / ネットワーク...etc.

# VCL データベース・アーキテクチャ

- VCL コンポーネントとして提供されるデータベース用クラス(コンポーネント)のセットからなる
  - 主に4つのレイヤー(層)に分けられる
    - 接続 / トランザクション
    - データセット
    - データソース
    - データコントロール
  - これらに加え、データモジュールが用意されている
    - ロジックを実装するための非 UI コンテナ
      - 開発時、非 UI コンポーネントをビジュアルに利用できる
      - データアクセス、制御、操作に関連するコンポーネントや操作ロジックを実装する
      - 積極的に利用することで、UI とロジックの分割を図ることができる



# VCL データベース・アーキテクチャ 接続 / トランザクション

- DB へ接続する形態毎に違う物が用意される
- DB への接続、およびトランザクションを管理する機能を持つ
  - ログイン / ログアウトの管理
  - 接続情報の管理
  - トランザクション制御
    - 機能によっては、トランザクション・コンポーネントと連携して実現
- TCustomConnection クラスから派生

# VCL データベース・アーキテクチャ

## データセット

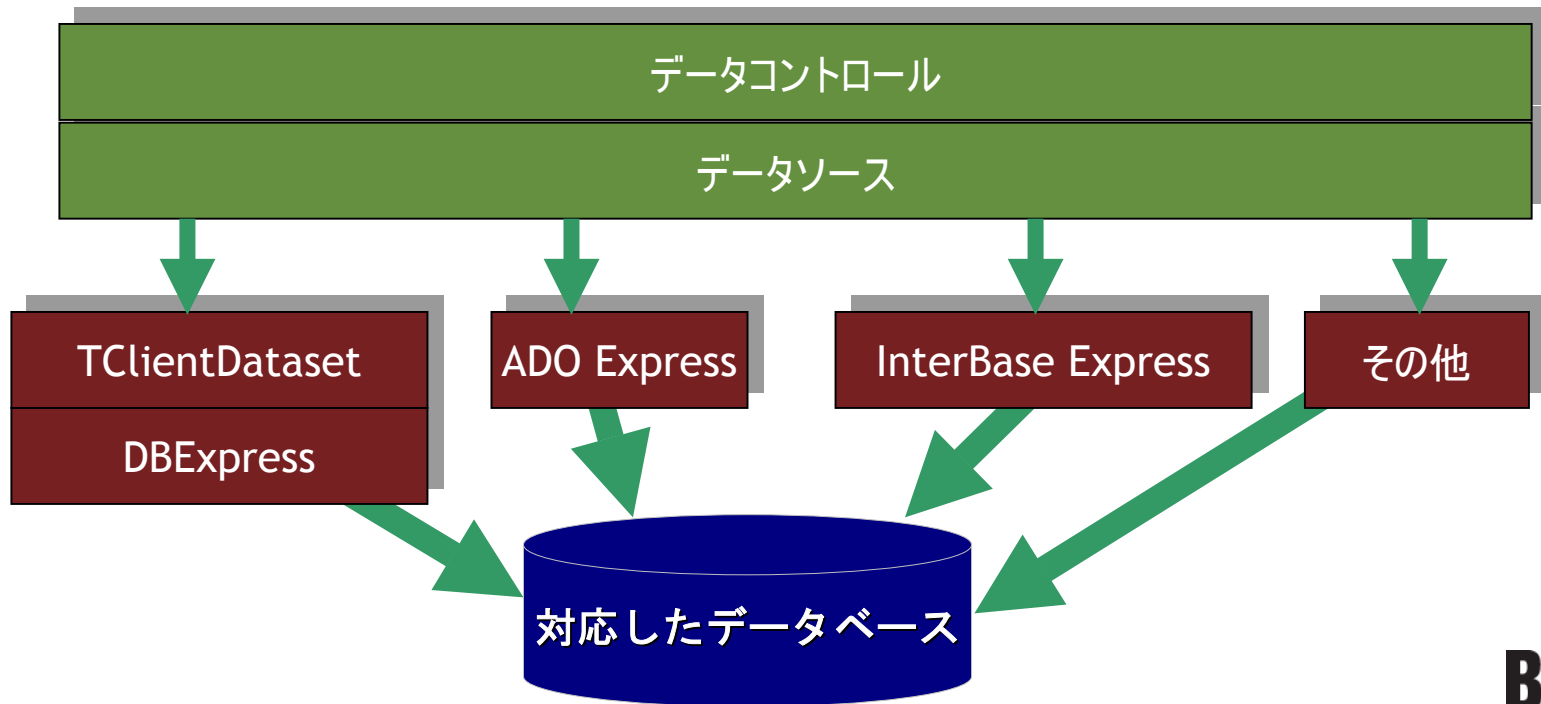
- DB へ接続する形態毎に違う物が用意される
- 主に機能毎に 3 種類の実装(クラス)が用意される
  - Table タイプのコンポーネント
    - 1つのテーブルやビューへのアクセス
  - Query タイプのコンポーネント
    - SQLによるデータアクセス
  - StoredProc タイプのコンポーネント
    - ストアド・プロシージャを実行する
  - それ以外の実装(クラス)が用意される場合もある
    - Dataset タイプのコンポーネント(複合型)
      - 上記 3 つの機能の複数を一つでカバーする
      - TDataSet ではなく、これから派生したコンポーネント
    - TClientDataSet
      - 2つの機能を持つ
      - 独立型ファイルベースデータセット
      - 別のデータセットにあるレコードのローカルなメモリ内バッファ
- TDataSet から派生

# VCL データベース・アーキテクチャ 接続 / トランザクション と データセット

- DB へ接続する形態毎に違う物が用意される
  - 汎用ミドルウェアに対応したもの
    - BDE(Borland Database Engine)
      - Borland が提供する高機能なデータアクセスミドルウェア
        - ファイル型 DB (Paradox / dBase / Microsoft Access) をサポート
        - ODBC を利用した接続にも利用可能
      - BDE コンポーネントセット
    - DBExpress
      - Borland が提供する最小機能のデータアクセスミドルウェア
      - DBExpress コンポーネント
      - TClientDataSet と共に利用する
    - ADO
      - ActiveX Data Objects
      - Windows に実装されている OLE DB のプログラミングインターフェース
      - ADO コンポーネントと共に利用する
  - データベースのクライアント API に依存したもの
    - InterBase Express
      - Borland InterBase 専用
  - その他 3rd Party 製品も存在する
    - Oracle Data Access Components
      - CoreLab 社 (<http://www.crlab.com/>) が提供
      - Oracle 社の Oracle DB 専用

# VCL データベース・アーキテクチャ 接続 / トランザクション と データセット

- DB へ接続する形態毎に違う物が用意される
  - 様々なものが用意されている
  - 自分で作成する事もできる
- 違いはデータセットまで、データソース、データコントロールは共通で利用する





# VCL データベース・アーキテクチャ 接続 / トランザクション と データセット

	BDE	DBExpress	ADOExpress	IBExpress
接続 / トランザクション	TDataBase	TSQLConnection	TADOConnection	TIBDatabase/TIBTransaction
データセット	Table タイプ	TTable	TADOTable	TIBTable
	Query タイプ	TQuery	TADOQuery	TIBQuery
	StoredProc タイプ	TStordProc	TADOStoredProc	TIBStoredProc
	Dataset タイプ	-	TSQLDataSet	TIBDataSet
データソース	TDataSource			
データコントロール	データコントロールコンポーネント			

## ■その他の違い

- DBExpress のデータセットは、TClientDataSet と連携させ、UIへのデータ表示や操作を実現させている
- IBExpress では Transaction をまかなうコンポーネントが独立している

# VCL データベース・アーキテクチャ

## データソースとデータコントロール

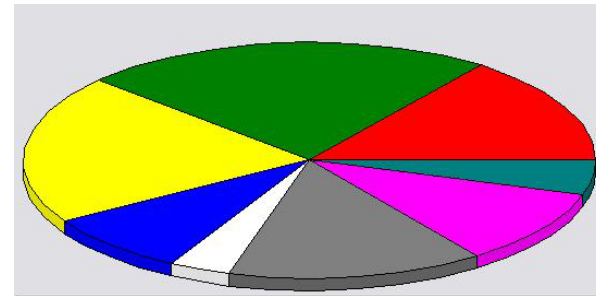
### ■ データソース

- データセットとデータコントロールとの間でのインターフェースを提供
- データセットとデータコントロールを互いに抽象化する
  - 多種の組み合わせを提供
- 各種データセットから共通に利用できる

EmpNo	LastName	FirstName	PhoneEx
2	Nelson	Roberto	250
4	Young	Bruce	233
5	Lambert	Kim	22
8	Johnson	Leslie	410
9	Forest	Phil	229
11	Weston	K. J.	34
12	Lee	Terri	256
14	Hall	Stewart	227
15	Young	Katherine	231
20	Papadopoulos	Chris	887
24	Fisher	Pete	888
28	Bennet	Ann	5
29	De Souza	Roger	288

### ■ データコントロール

- データを表示、操作する UI 部品
  - グリッド、エディットボックスだけではなく、グラフなども存在

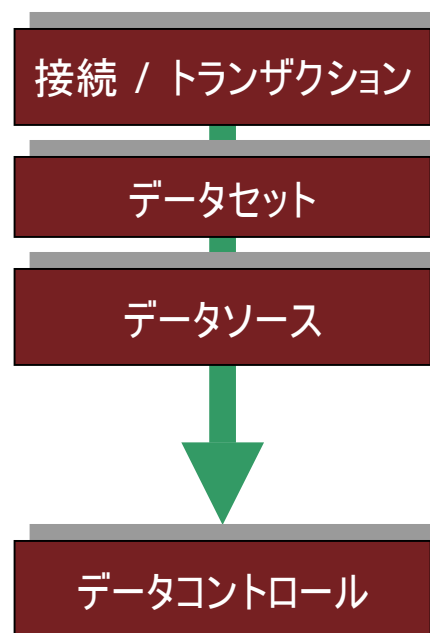


# VCL データベースアーキテクチャ

- データアクセス部分とデータ表示部分が分離されている
  - 様々な入力と出力の組み合わせが可能
    - 入力
      - 様々なデータベースからの入力
        - InterBase / ORACLE / 他、対応する RDBMSや、DBファイル
        - TDataset を活用する事で、標準で対応していなオリジナルの実装も可能
    - 出力
      - 分散オブジェクトクライアント
      - Web サーバーアプリケーションにより Web ブラウザへ
      - Web Service により、遠隔地のシン・クライアントへ
      - OLE を活用し、EXCEL や 他アプリケーションへ

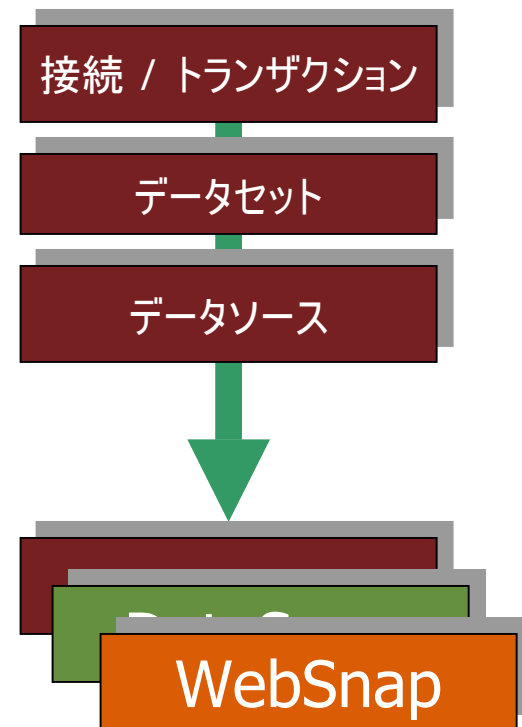
# VCL データベース・アーキテクチャ

- VCL ではこの4つの層のコンポーネントを連携させて DB 周りの実装を行なう
  - 連携はプロパティ指定で行なう
  - プログラマブルに変更可能
- これらの層の役割と違いを理解することで、効果的な開発を行える
  - 接続 / トランザクション
  - データセット
  - データソース
  - データコントロール



# VCL “Snap” アーキテクチャ

- VCL データベースアーキテクチャと連携し、データベースロジックをさらに活用できるように拡張できる
  - DataSnap
    - データセットとデータコントロールの間に入り、リモートでこれらを連結できるように拡張する。
    - 簡易ではあるが多層 C/S を実現し、シンクライアントを実現する
  - WebSnap
    - Web アプリケーション開発をサポート
    - データセットを利用し、Web アプリケーションのデータロジックを作成



# この章のまとめ

- Borland Developer Studio (Delphi / C++Builder) ではビジュアル開発の結果、再利用性の高いコードが生成される
  - その仕組みを理解した上でビジュアル開発の機能を利用する
  - 単純に、ビジュアルの利便性だけで開発を行なってはいけない
- ビジュアル開発では豊富なコンポーネントが利用できる
  - VCL フレームワークの約束事にしたがって作成された物
    - Borland が提供する物も独自に開発した物も扱いは等価
- データベース系のコンポーネントが多数用意されている
  - 大別するとデータを操作するコンポーネントとコントロールするコンポーネントに分かれる
    - データを操作するコンポーネントは接続種別で違う物を利用するが、データをコントロールするコンポーネントは同じ物を利用できる