

embarcadero

17th Embarcadero
Developer Camp

【A3】 Delphi/C++テクニカル セッション
「旧Delphiで作成された
アプリケーションを
Delphi 2010に移行するには」

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



embarcadero

17th Embarcadero
Developer Camp



はじめに





スピーカープロフィール

17th Embarcadero
Developer Camp

山本 悟

::略歴::

17歳からコンピュータ関係にはまり、プログラミング言語で一番最初に触ったのはTurbo C++でした。大学卒業後、そのまま起業しましたが、今でも自分勝手に創るのが一番好きです。

現在の開発案件では、主に.NETとPHPを利用しています。

Delphiとは1.0で出会い、バイト先のゲーム会社のサポートプログラムを作成したりしていました。

現在も、ちょっとしたプログラムはDelphiで作ることが多く、最後のスライドで紹介するDH-APPLIももちろんDelphi製です。

現在はPM系や言語系といったIT系のトレーニングを多くやっていますが、昔テレビ埼玉でパソコンの先生をしていたこともあります。

株式会社ドリームハイブ	: http://www.dreamhive.co.jp/
フリーソフトなどの配布場所	: http://dhive.jp/
ブログ	: http://dhive.jp/blog/yama/
facebook	: http://www.facebook.com/kryu2
mixi	: http://mixi.jp/show_friend.pl?id=104357

- 移行の基準を考える
- 移行の手順について
- 移行の際のポイント
- まとめ

- このセッションは、テクニカル・セッションです
- Delphi for Win32のアプリケーションの移行についてお話しいたします
- すべての移行点を網羅するのではなく、私の主観と経験に基づいて構成しています
- 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

embarcadero

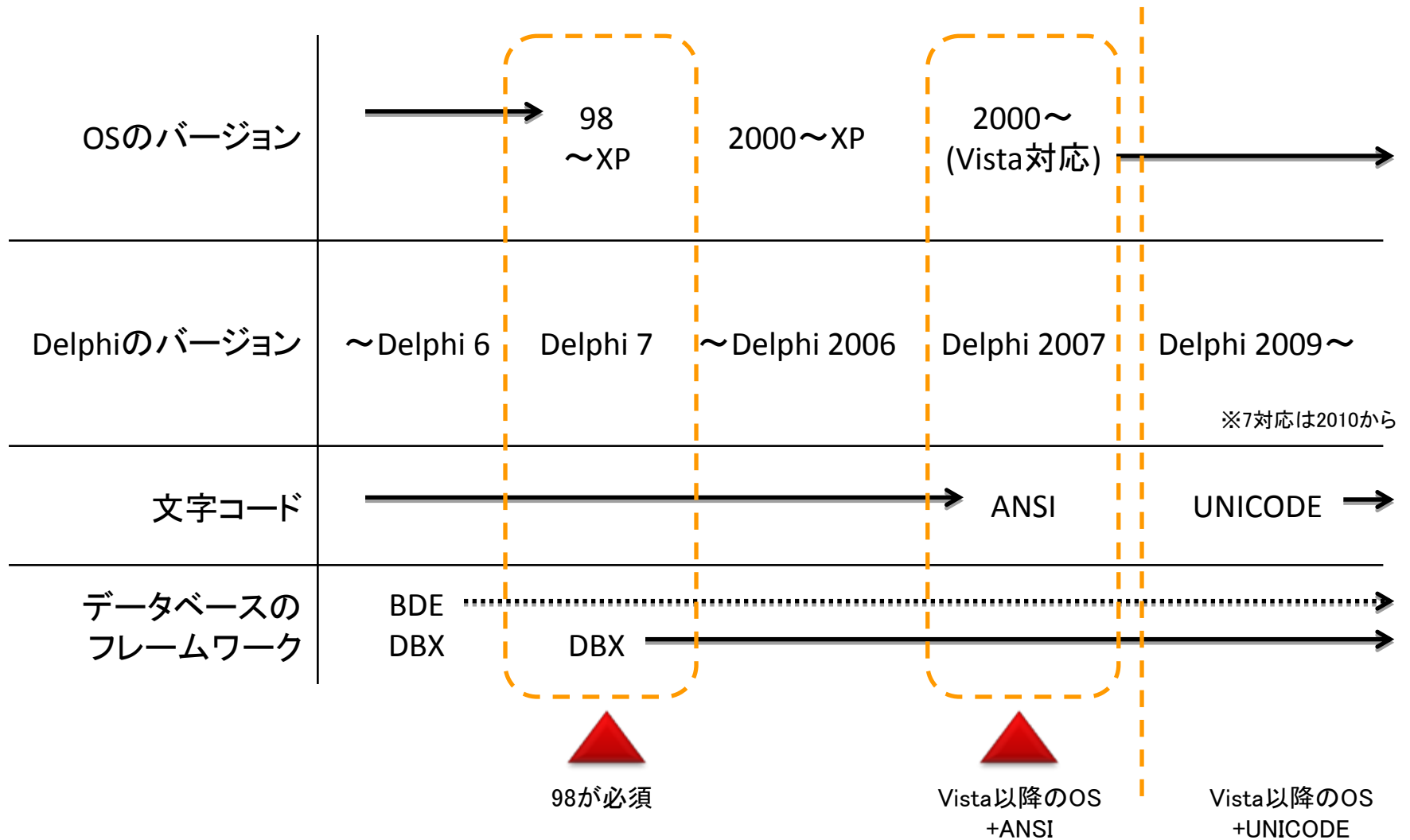
17th Embarcadero
Developer Camp



移行の基準を考える



移行の基準を考える

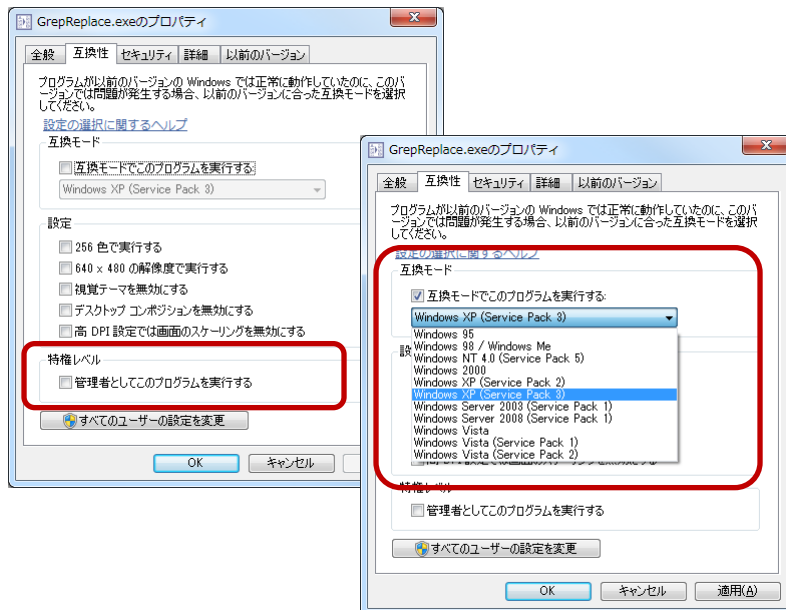


とりあえず、だいたい動きます\(^_^)/

- プログラムの「互換性」タブを使用するとか

- 新しいDelphiに切り替えるだけでも効果あります

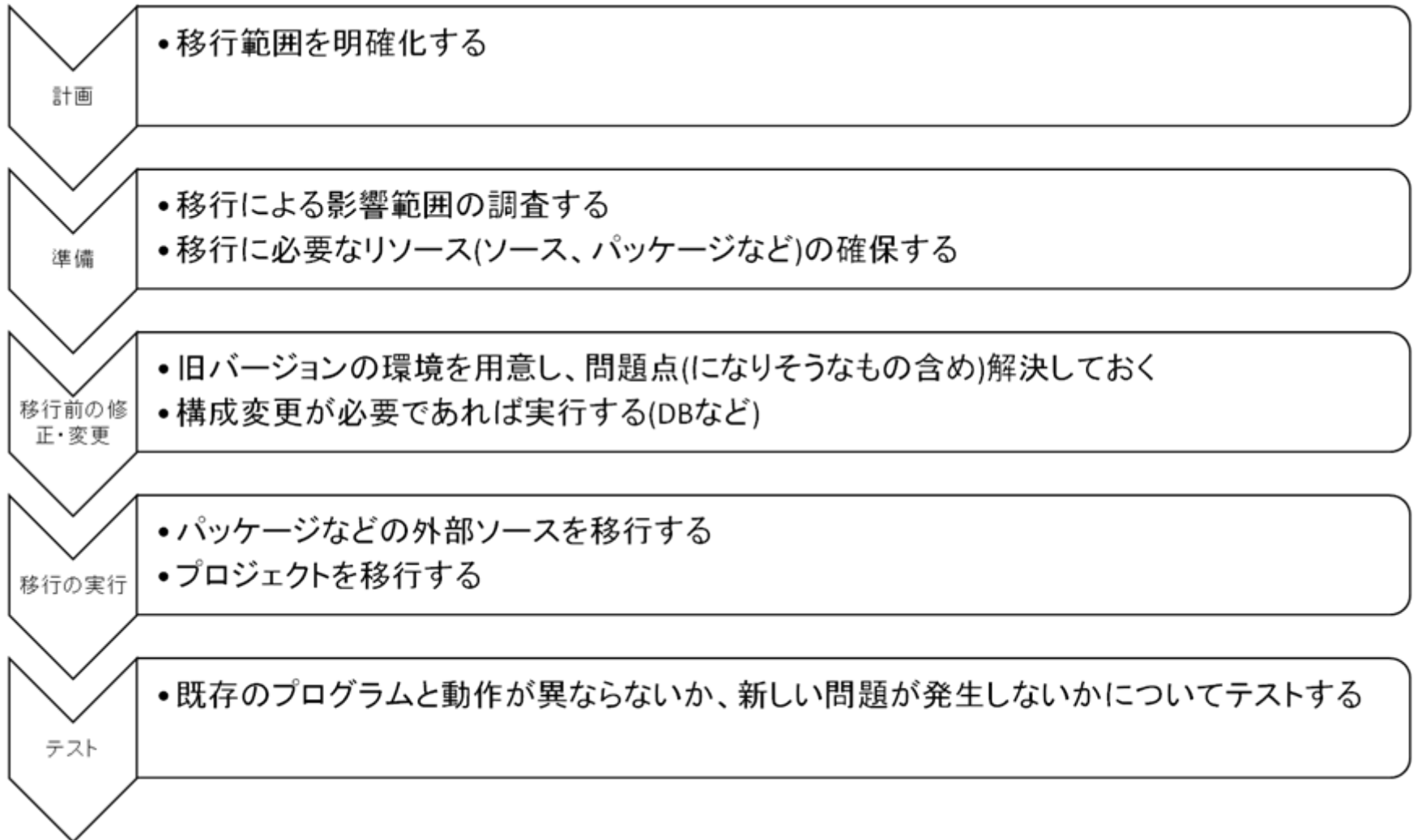
- 新しいコンパイラによる効果(最適化や新機能が有効になる)があります
- ドライバ・パッケージ・ソースなどが更新されています





移行の手順について





- とりあえず新しいDelphiで開いてみましょう

ただし、規模の大小に関わらず、

- Delphi 2007より前のプロジェクトについては
一度Delphi 2007へ移行してから、
最新のDelphiに移行することを強くオススメします

embarcadero

17th Embarcadero
Developer Camp



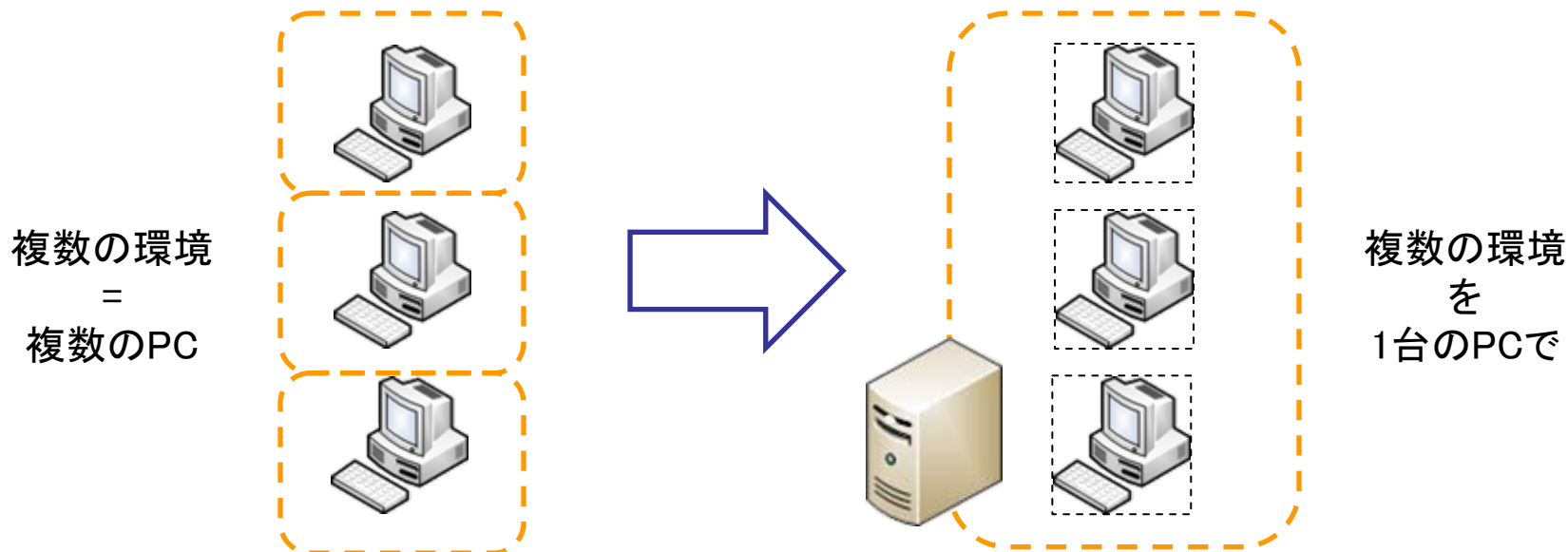
移行の際のポイント



1. 開発環境はどうするのが良い？
2. 移行範囲を分けて考えましょう
 - 管理者権限関係
 - UI関係
 - DB関係
 - その他
3. 移行しやすい設計をいくつか紹介

- 仮想環境を構築するのがオススメ

- 仮想化: 1台のコンピュータを、あたかも複数台のコンピュータであるかのように論理的に分割し、それぞれに別のOSを動作させることで、複数の環境を少ないリソースで比較的安全に構築することができる
 - *Hyper-V*: マイクロソフト
 - *VMWare*: ヴィエムウェア
 - *Xen*: シトリックス・システムズ・ジャパン など



2. 管理者権限関係

- Vista以降、管理者権限で動作させることを前提としたアプリケーションで問題が発生します。
- UAC(ユーザーアカウント制御)

問題が発生する処理

- 「Program Files」フォルダ内のファイルの変更
- 「Windows」や「System32」フォルダ内のファイルの変更
- 「HKLM¥Software」以下にあるレジストリの変更
- ローカルコンピュータの日時の変更
- 「サービス」のインストール・アンインストール
- ..

対策の方針

- UACが存在する前提で動作するようにアプリケーションを修正する
 - 標準ユーザー権限で動作するように作る
 - UACでも動作するように、管理者権限が必要なコードは別のexeなどにする
 - マニフェストファイルでUAC権限レベルを指定する
- 実行時に対処する
 - アプリケーションが権限を昇格できるようにする
 - ファイルやフォルダに対するACL(アクセス制御リスト)を設定する

<<仕様を次の様に変更することで対応>>

- 「Program Files」フォルダ内のファイルの変更

- 「C:¥ProgramData」や「%UserProfile%」にファイルを配置すべきですが、SHGetFolderPath関数などにCSIDL定数を引数として渡し、利用可能なフォルダパスを取得するのが良い
- 不必要に、書き込み権限付きでオープンしない

- 「HKLM¥Software」以下にあるレジストリの変更

- 「HKCU」への書き込みに変更する

- テーマサポート
 - 「プロジェクト|オプション」メニュー、「アプリケーション」ツリーを選択後、「アプリケーションの設定」の「ランタイムテーマを有効にする」チェックボックスをオンにする
- デフォルトフォントとフォントサイズ
 - XPではTahoma 8、Vistaではメイリオ 9であり、DelphiのデフォルトのCHARSETも違う。Delphi 2009以降でサロゲートペアや字形の違う文字を使ったアプリを作成し、XPIに持って行くと化けるので、MS UIゴシックなどを指定しておく
ParentFontプロパティをTrueにするのを忘れずに
 - ただし、検証が面倒ならいじらないのもアリ！！
- ComboBoxでバックスペースすると全角文字が化ける
 - AutoCompleteプロパティをFalseにすることで回避できる
- アイコンサイズ
 - XPまでは16x16、32x32だったが、Vistaからは48x48が必要、さらには256x256まで利用することが可能になった
 - アルファチャンネル、256x256サイズをサポートするのはPNG形式
 - 私はIcoFXで作っています → <http://icofx.ro/>
- タスクバーでの表示
 - ShowMainFormOnTaskBar := Trueにする
- ウィンドウの半透明化
 - GlassFrameプロパティを設定する
- 新しいダイアログ
 - TOpenDialog/TSaveDialogなどは、UseLatestCommonDialogsグローバル変数と合わせて使う
 - TaskDialogは、標準のTaskMessageDlgを使うのでも良いが、
 - TMS SoftwareのTAdvTaskDialogを使うと多機能な上に楽 → <http://www.tmssoftware.com/site/vtd.asp>

<<DBの移行は2パターンしかない>>

●BDE系で頑張る

- Delphi 7付属のBDE5.2で開発およびメンテナンスが終了してるので、正規サポートは望めない(現在附属しているBDEもDelphi 7 Professionalレベルのものであり、EnterpriseエディションのSQLLinkは含まれていない)
- あくまでも、過去の資産(dBase/Paradox,ODBC接続)の保守用
- 3rdパーティ製の互換コンポーネントを利用する
- どちらにしても、開発者の責任が大きい

●DBXへ移行する

- dbExpressへの切り替えを行う
- 修正範囲が元の作りによっては非常に大きい

- VCLのライブラリ構成の変更
 - クラス継承、ユニット名の変更
 - クラスの定義が別ユニットへ移動
 - 問題に遭遇した場合は、新しいユニットなどへのパスを指定する
 - 「未定義の識別子」コンパイルエラーなどが発生します
- ヘルプ
 - WinHelpが標準ではなくなった
 - Microsoftが提供するVista/7用のWinHelpモジュールをインストールしてもらおう
- レポートツール
 - QuickReportは添付されず、RaveReportは日本語に難あり
 - 手間をかけたくないのであれば、QuickReportの有償版を購入する
 - Fast Reportがオススメ
 - <http://fast-report.com/>

3. 移行しやすい設計をいくつか

- 移行しやすいとは

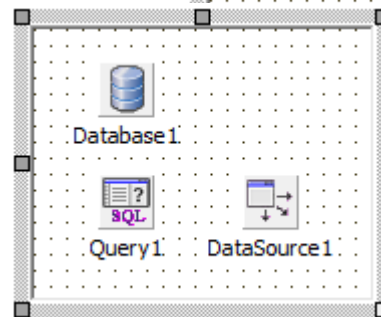
- 変更点が少なく、移行後のテスト範囲が明確になっていること！

- 設計の例

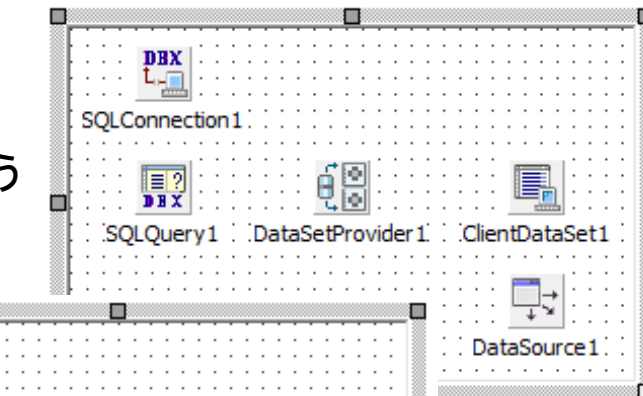
- データモジュールの利用
- クラスだけでなく、例外処理などもカプセル化してしまう
- BDE+ClientDataSetへの変更作業を中間に挟む
- メソッド内でオブジェクト参照の付け替え

```
var  
  MyMemo: TMemo;  
begin  
  MyMemo := Memo1;  
  
  MyMemo.Lines.Add('AAA');  
  MyMemo.Lines.Add('BBB');  
  MyMemo.Lines.Add('CCC');  
end;
```

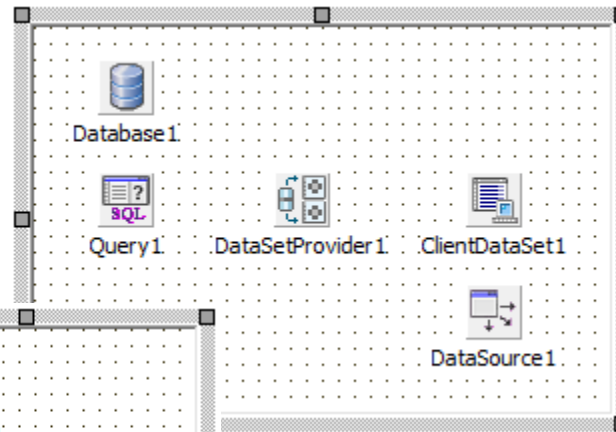
BDE構成の
データモジュール



DBX構成の
データモジュール



BDE+ClientDataSet構成の
データモジュール



embarcadero

17th Embarcadero
Developer Camp



まとめ



- 移行範囲は明確にすること
 - 特にデータベースを利用したアプリケーションでは、BDE/DBXなどのフレームワークの違いによって大きく工数が変わります
- 予めソースを触れる環境にあるのであれば、移行のための準備としてリファクタリングしておくこと
 - メンテナンス性が低いほど、移行の工数がかかります
 - Delphi 2005以降、リファクタリングのための機能が数多くIDEに統合されています
- 複数のバージョンがテストできる環境を用意すること
 - ここ数年で、仮想環境構築のコストは極端に下がっています
 - 実機よりも仮想環境! 環境依存の問題検証をする人件費よりも安いです

移行のし易さは、設計に強く依存します!

embarcadero

17th Embarcadero
Developer Camp



Q & A



ありがとうございました