

【セッションNo. 2】

Delphi/400開発ノウハウお教えします

「バージョンアップのポイント & Webアプリ開発実演」

株式会社ミガロ.

システム事業部 システム3課

小杉 智昭

【アジェンダ】

バージョンアップのポイント

- 1) バージョンの変遷
- 2) バージョンアップを推奨する理由
- 3) バージョンアップ時の修正ポイント

Webアプリ開発実演

- 4) Delphi/400とWebアプリ開発
- 5) Webアプリ開発① - Ajaxの利用
- 6) Webアプリ開発② - Webサービスの利用

バージョンアップのポイント

■ 1) バージョンの変遷

● はじめに

2009年9月にDelphi/400 Ver2009がリリースされ、本格的にUnicode対応が行われました。

また、Windows XP対応やWindows Vista対応も必要となる機会が増えていることと思います。

このセッションの前半では、今回はこれまでリリースされてきたDelphi/400のバージョンによる違いを確認し、バージョンアップのポイントをご紹介します。

■ 1) バージョンの変遷

● バージョンによる違い

バージョン	5	6	7
発売年	1999年	2001年	2002年
動作環境	Win95、98/SE WinNT4.0	Win98/SE、Me WinNT4.0、2000	Win98/SE Win 2000、XP
文字コード	SJIS	SJIS	SJIS
データベース	BDE 5	BDE 5 dbExpress(DBX)	BDE 5 dbExpress(DBX)
Web	WebBroker	WebSnap	IntraWeb 5

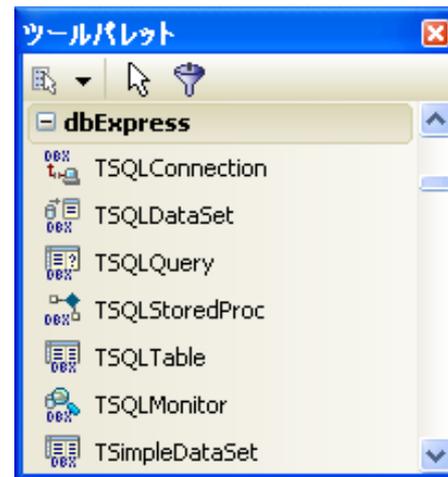
バージョン	2005	2006	2007	2009
発売年	2004年	2005年	2007年	2008年
動作環境	Win 2000、XP Win Server 2003	Win 2000、XP Win Server 2003	Win 2000、XP、Vista Win Server 2003	Win 2000、XP、Vista Win Server 2003、2008
文字コード	SJIS	SJIS	SJIS	Unicode
データベース	BDE 5 dbExpress(DBX)	BDE 5 dbExpress(DBX)	BDE 5 dbExpress(DBX4)	BDE 5 dbExpress(DBX4)
Web	IntraWeb 7	IntraWeb 8	VCL for the Web 9	VCL for the Web 10

■ 2) バージョンアップを推奨する理由

● dbExpress接続方式の追加

Delphi/400 V6 から従来のBDE接続方式に加えて、dbExpress接続方式が追加され、接続可能なデータベースが増えただけでなく、プログラムの幅が広がりました。

Delphi/400 V2007からは更にdbExpress接続方式の見直しが図られ、旧dbExpressコンポーネントとの互換性を保ちつつ、プーリング機能等の追加が行われました。



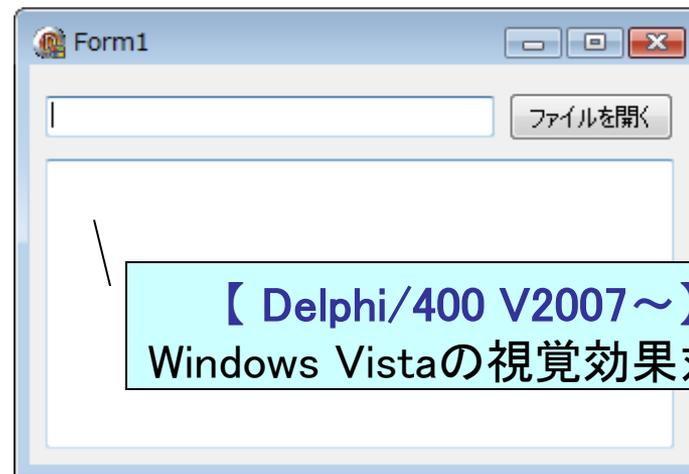
■ 2) バージョンアップを推奨する理由

● 新OS対応

Windowsの新バージョン対応は随時行われていますが、外観で特に大きな影響があったWindows XPのテーマ対応、Windows Vistaの視覚効果対応が行われています。また次期V2010ではWindows7が正式対応予定です。



【 Delphi/400 V7～ 】
Windows XPのテーマ対応

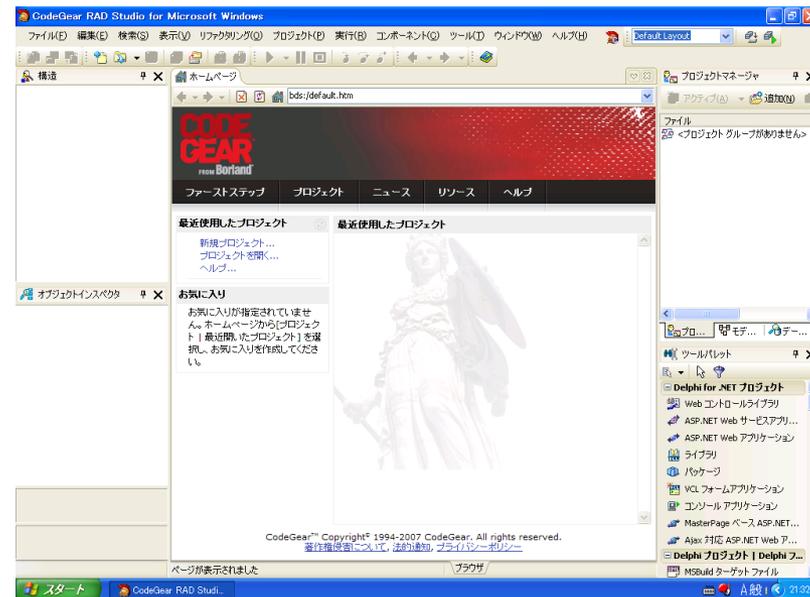
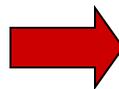
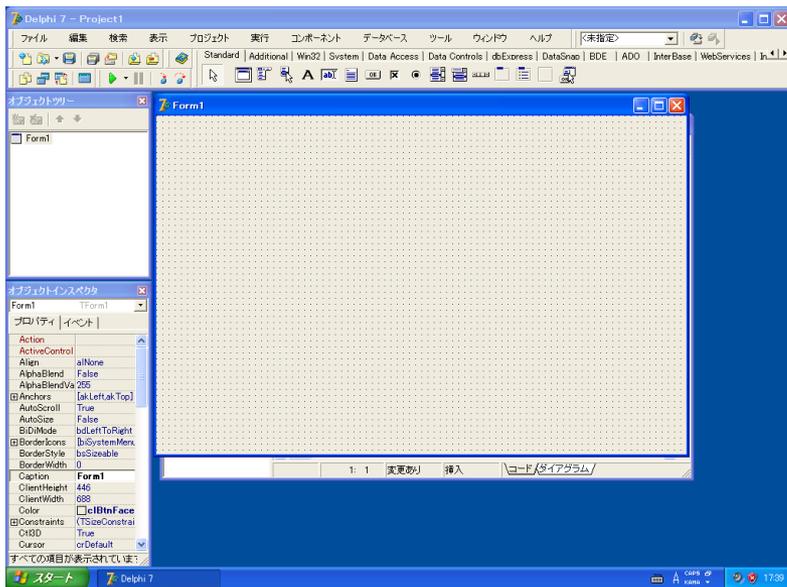


【 Delphi/400 V2007～ 】
Windows Vistaの視覚効果対応

■ 2) バージョンアップを推奨する理由

● 統合開発環境の強化

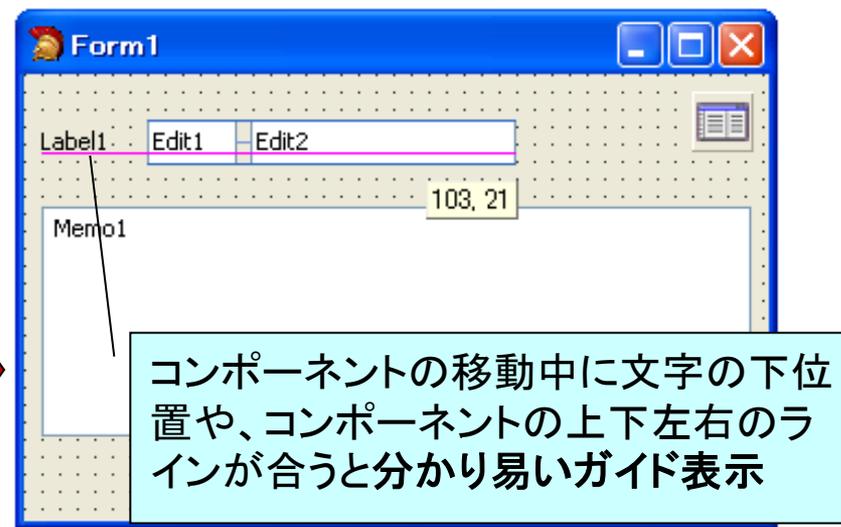
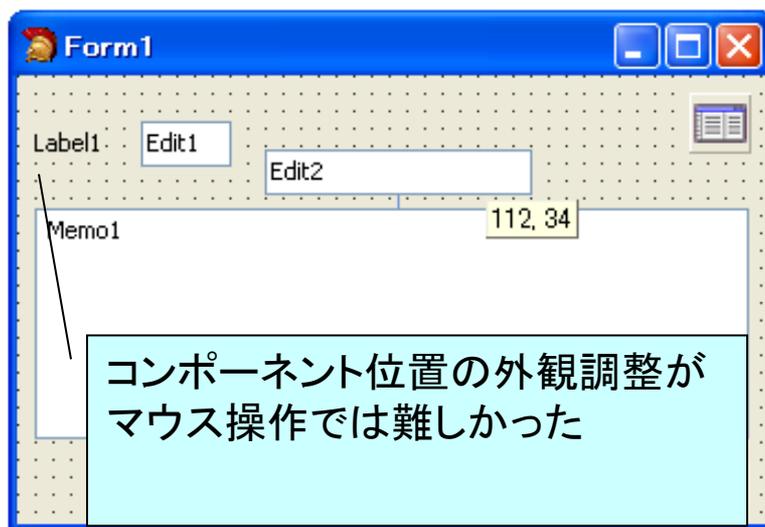
統合開発環境 (IDE) が一新され、多くの機能が追加されています。
(参考) <http://edn.embarcadero.com/jp/article/34361>



■ 2) バージョンアップを推奨する理由

● 統合開発環境の強化(強化機能ピックアップ①)

Delphi/400 V2006～デザインガイド機能の追加



■ 2) バージョンアップを推奨する理由

● 統合開発環境の強化(強化機能ピックアップ②)

Delphi/400 V2007～ライブテンプレート機能の追加

```
procedure TForm1.Button1Click(Sender: TObject);
begin
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  for
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  for iLoop := 0 to Table1.Fields.Count - 1 do
  begin
end;

procedure TForm1.Button1Click(Sender: TObject);
var
  iLoop: Integer;
begin
  for iLoop := 0 to Table1.Fields.Count - 1 do
  begin
    |
  end;
end;

procedure TForm1.Button1Click(Sender: TObject);
var
  iLoop: Integer;
begin
  for iLoop := 0 to Table1.Fields.Count - 1 do
  begin
    |
  end;
end;
```


■ 2) バージョンアップを推奨する理由

● 統合開発環境の強化(強化機能ピックアップ④)

Delphi/400 V2006～ソース変更バーの追加

```
procedure TForm1.Button1Click(Sender: TObject);
var
30  iLoop: Integer;
begin
  Memo1.Lines.Clear;
  for iLoop := 0 to Table1.Fields.Count - 1 do
  begin
    Memo1.Lines.Add(Table1.Fields[iLoop].FieldName);
  end;
  Memo1.SetFocus;
38  Memo1.SelectAll;
end;
40
```

変更後、保存済の箇所(黄緑色)

変更後、未保存の箇所(黄色)

■ 2) バージョンアップを推奨する理由

- 統合開発環境の強化(強化機能ピックアップ⑤)

Delphi/400 V2005～行番号表示の追加

The screenshot shows the CodeGear RAD Studio for Microsoft Windows interface. The menu bar includes options like 集(E), 検索(S), 表示(V), リファクタリング(O), プロジェクト(P), and 実行(R). The '検索(S)' menu is open, showing options such as 検索(E)... (Ctrl+F), ファイル検索(D)... (Shift+Ctrl+F), 置換(R)... (Ctrl+R), 次を検索(S) (F3), インクリメンタルサーチ(I) (Ctrl+E), クラスの検索(C)..., リファレンスの検索(Z), ローカルリファレンスの検索(L) (Shift+Ctrl+Enter), オリジナルシンボルの検索(O) (Ctrl+G), 指定行に移動(G)... (Alt+G), and 指定アドレスに移動(E). The '指定行に移動(G)...' option is highlighted. A dialog box titled '指定行に移動' is open, showing a text field for '移動先の行番号(E):' with the value '38' and buttons for 'OK', 'キャンセル', and 'ヘルプ'. The background shows a code editor with a table definition and a procedure definition. A yellow highlight is under the procedure definition, and a green highlight is under the table definition. A blue box highlights the '行毎)' label, and another blue box highlights the 'ソル行)' label.

■ 2) バージョンアップを推奨する理由

● .NET開発環境対応

Delphi2005からは従来のWin32環境のみでなく、.NET環境での開発が可能になり、Delphi/400もV2006で.NETに対応しました。

CUSTNO	COMPANY	ADDR1	ADDR2
1221	ココナッツマリッジショップ	大島町4-976-321	
1231	アクアダイビングセンター	明太区曾根541	
1351	亀山ダイブセンター	稲毛区亀山町632-1	
1356	ダイビングハウスサーフ	松島町7-737	
1380	ダイブショップブルーリーフ	鯖松町23-738	
1384	MHMダイバーズクラブ	埴輪町32	
1513	ダイブハウススタート	東荻5-8-7	
1984	ADVENTURE UNDERSEA	PO BOX 64594	
2118	グリーンスポーツクラブ	中海老町633-21	東渡辺町3-147
2135	パynaッブルダイバーズ	中村1455-1	
2156	マリッジハウスペンギン		
2163	SCUBA HEAVEN		
2165	SHANGRI-LA SPORT		
2315	ダイビングショップマーメイド	富ヶ谷3-21-4	
2353	マリッジセンター 築楠浦	西区築楠浦5-6-11	

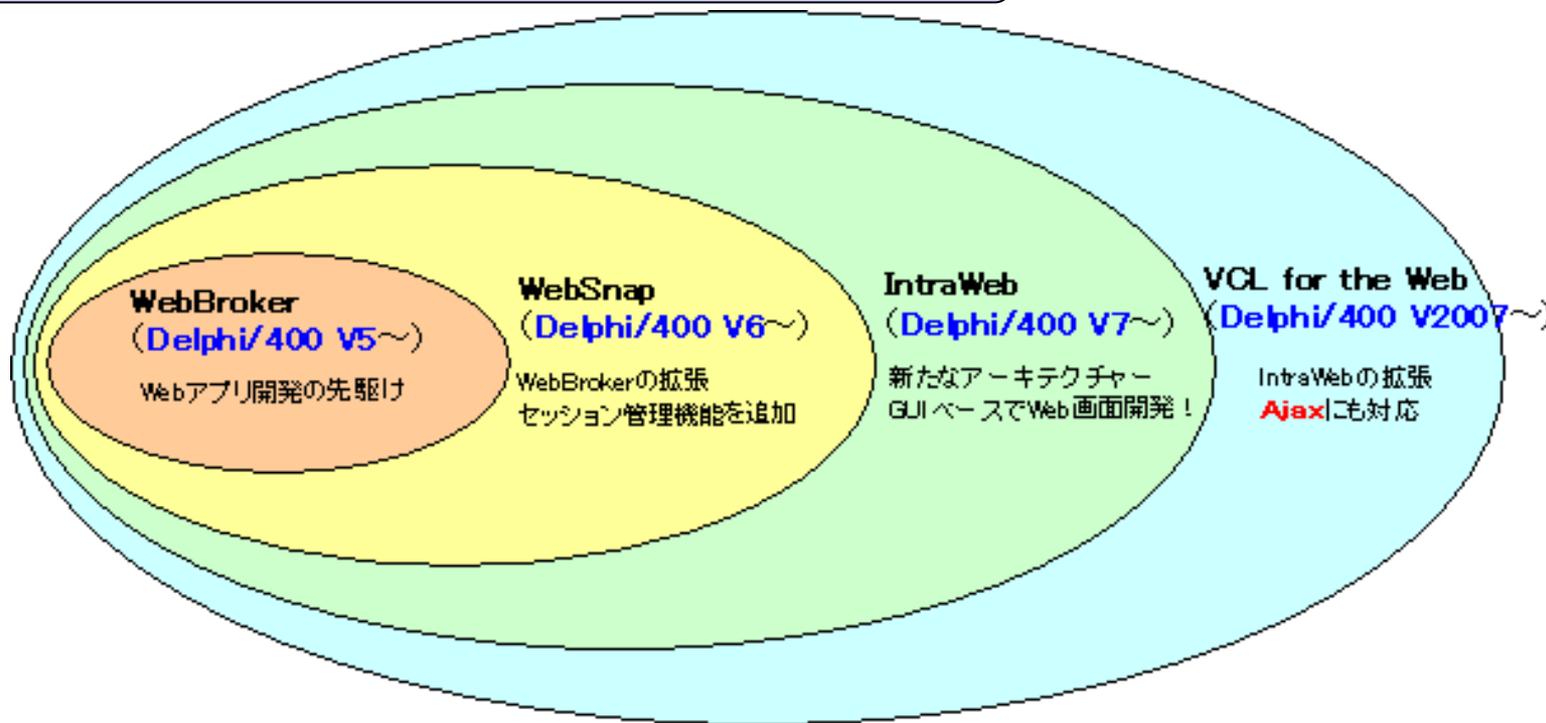
通常のwin32アプリケーション

.NETアプリケーションも同様に開発可能

■ 2) バージョンアップを推奨する理由

● Web開発機能の強化

WebBrokerからVCL for the Webへ



■ 2) バージョンアップを推奨する理由

● Unicode対応

Delphi/400 V2009からは標準文字セットがUnicode対応

プログラム内で扱われる文字コードの標準がANSI(Shift-JIS)からUnicode(UTF-16)へ変更になり、プログラム内やコンポーネント上で多彩な文字が取り扱えるようになりました。

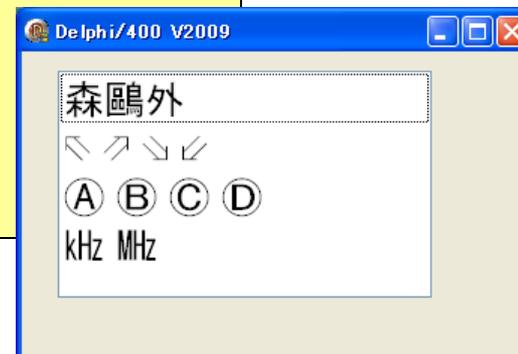
【利用可能となる文字の例】

人名漢字(森鷗外の「鷗」※通常は「鷗」で代用)

特殊記号(↖ ↗ ↘ ↙)

丸付き文字(Ⓐ Ⓑ Ⓒ Ⓓ)

単位記号(kHz MHz)



■ 3) バージョンアップ時の修正ポイント

● バージョンアップを行うにあたって

Delphi/400のバージョンアップの基本は『リコンパイル』



Delphi/400 は非常に互換性が高い開発環境になっており、基本的にはソースコードを新バージョン環境で読み込んで、リコンパイルすればほぼそのままバージョンアップできます。

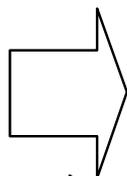
しかし、プログラムの内容によっては例外的に修正が必要となるケースがありますので、そういったケースと対応方法についてご紹介していきます。

■ 3) バージョンアップ時の修正ポイント

● バージョンアップを行う際のキーポイント

ポイントA

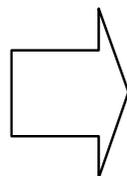
:
Delphi/400 V5



Delphi/400 V6
Delphi/400 V7

ユニット再編
設計用コードの分離

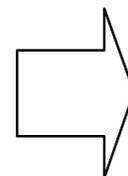
ポイントB



Delphi/400 V2005
Delphi/400 V2006
Delphi/400 V2007

統合開発環境の強化
Developer Studio化

ポイントC



Delphi/400 V2009
:

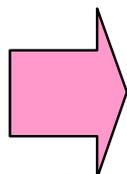
ユニコード対応

■ 3) バージョンアップ時の修正ポイント

● ポイントAに関する注意点

ポイントA

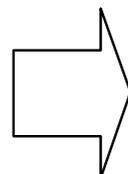
:
Delphi/400 V5



Delphi/400 V6
Delphi/400 V7

ユニット再編
設計用コードの分離

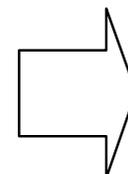
ポイントB



Delphi/400 V2005
Delphi/400 V2006
Delphi/400 V2007

統合開発環境の強化
Developer Studio化

ポイントC



Delphi/400 V2009
:

ユニコード対応

■ 3) バージョンアップ時の修正ポイント

● ポイントA (Delphi/400 V5以前からDelphi/400 V6以降)

ユニット再編

Delphi/400 V6の段階でCLX対応のために、ソースコードの再編が行われました。このポイントをまたぐバージョンアップを行う場合、uses節の修正がよく必要になります。

uses節の初期状態 (Delphi/400 V5)

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs;

uses節の初期状態 (Delphi/400 V7)

uses

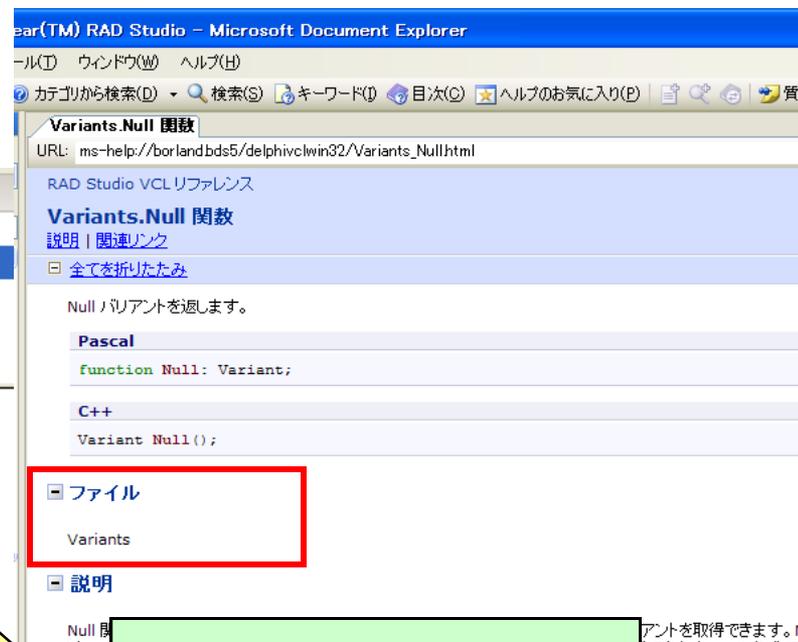
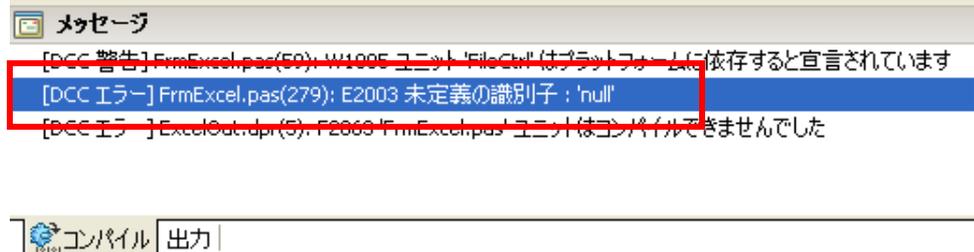
Windows, Messages, SysUtils, **Variants**, Classes, Graphics, Controls, Forms, Dialogs;

新しいユニットの追加

3) バージョンアップ時の修正ポイント

- ポイントA (Delphi/400 V5以前からDelphi/400 V6以降)

ユニット再編



未定義の識別子: 'xxxx'

ヘルプでユニットを確認

'xxxx' が定義されている
ユニットをuses節に追加

■ 3) バージョンアップ時の修正ポイント

● ポイントA (Delphi/400 V5以前からDelphi/400 V6以降)

設計用コードの分離

Delphi/400 V6からコンポーネントのソースコードで設計時と実行時のコードを明確に分離しなければならなくなりました。同時にデザインインターフェースユニットのユニット名も変更されました。

対象

uses節に**DsgnIntf**を含むDelphi/400 V5以前のコンポーネントソースコード

対処法

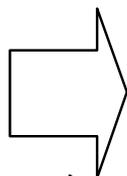
uses節の**DsgnIntf**を**DesignIntf**へ変更し、**DesignEditors**, **VCLEditors**, **RTLConsts**を追加し、パッケージのrequiresリストに**designide**を追加します。

■ 3) バージョンアップ時の修正ポイント

● ポイントBに関する注意点

ポイントA

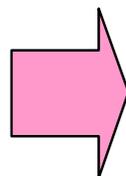
:
Delphi/400 V5



Delphi/400 V6
Delphi/400 V7

ユニット再編
設計用コードの分離

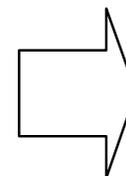
ポイントB



Delphi/400 V2005
Delphi/400 V2006
Delphi/400 V2007

統合開発環境の強化
Developer Studio化

ポイントC



Delphi/400 V2009
:

ユニコード対応

■ 3) バージョンアップ時の修正ポイント

- ポイントB (Delphi/400 V7以前からDelphi/400 V2005以降)

フォームの初期表示

統合開発環境内でのフォームの表示方法が変更され、
フォームの初期表示される位置が変更されました。

Delphi/400 V7以前のPositionプロパティ

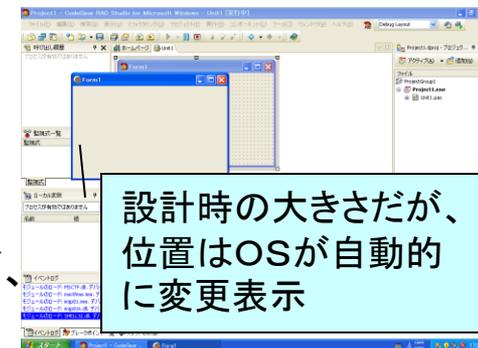
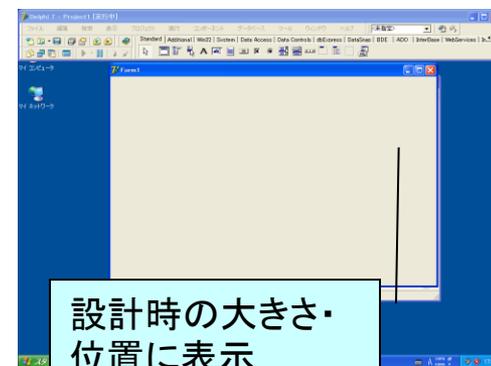
【poDesigned】

フォームは設計時と同じ画面上の位置、高さ、幅で表示

Delphi/400 V2005以降のPositionプロパティ

【poDefaultPosOnly】

フォームは設計時にフォームを作成したサイズで表示されるが、
画面上の位置はオペレーティングシステムが選択



■ 3) バージョンアップ時の修正ポイント

- ポイントB (Delphi/400 V7以前からDelphi/400 V2005以降)

Fontプロパティのデフォルト値

Fontプロパティのデフォルト値が変更されたため、フォントを指定していない場合は見た目が異なります。

Delphi/400 V7以前のFontプロパティ

フォント名: **MS P ゴシック** フォントサイズ: **9**
 CharSet: **SHIFTJIS_CHARSET**

Delphi/400 V2005以降のFontプロパティ

フォント名: **Tahoma** フォントサイズ: **8**
 CharSet: **DEFAULT_CHARSET**

V7までのデフォルト値

MS P ゴシック、9ポイント

abcABCあいう

V2005からのデフォルト値

Tahoma、8ポイント

abcABCあいう

V7以前で作成したプロパティ値が設計画面で引き継がれたコンポーネントは問題ありませんが、動的に作成しているコンポーネントなどは注意してください。

■ 3) バージョンアップ時の修正ポイント

- ポイントB (Delphi/400 V2006以前からDelphi/400 V2007以降)

Transparentプロパティのデフォルト値

V2006までのTransparentプロパティ

False

V2007からのTransparentプロパティ

True

Labelコンポーネントの例

V2006まではデフォルト値がFalse

得意先コード



V2007からはデフォルト値がTrue

得意先コード

Falseに設定すれば
本来の表示になります

Transparentプロパティのデフォルト値はフォーム情報(dfm)に保存されません。コンパイルするDelphi/400のバージョンで値が変わってしまうので注意してください。

3) バージョンアップ時の修正ポイント

- ポイントB (Delphi/400 V7以前からDelphi/400 V2005以降)

開発対象環境の選択

初回の読込時にWin32用か.NET用へアップグレードするかの確認ダイアログが表示されることがあります。通常はWin32側を選択して下さい。



■ 3) バージョンアップ時の修正ポイント

- ポイントB (Delphi/400 V2005以前からDelphi/400 V2006以降)

dbExpress

Delphi/400 V2006からCO400Connectionで使用するdllがdbco400.dllから新しいdbco430.dllに変更されています。



LibraryName
VendorLib
を
V2005以前:dbco400.dll
V2006以降:dbco430.dll
で指定してください

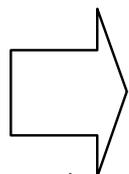
またV2007以降でトランザクション処理を行う場合、IsolationLevelの指定を“BlobSize”で指定する必要があります。
<http://www.migaro.co.jp/contents/products/delphi400/faq/tec.html#q50>

■ 3) バージョンアップ時の修正ポイント

● ポイントCに関する注意点

ポイントA

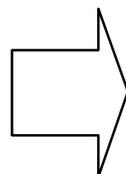
:
Delphi/400 V5



Delphi/400 V6
Delphi/400 V7

ユニット再編
設計用コードの分離

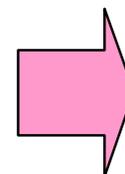
ポイントB



Delphi/400 V2005
Delphi/400 V2006
Delphi/400 V2007

統合開発環境の強化
Developer Studio化

ポイントC



Delphi/400 V2009
:

ユニコード対応

■ 3) バージョンアップ時の修正ポイント

- ポイントC (Delphi/400 V2007以前からDelphi/400 V2009以降)

文字列長の取り扱いの違い

IBM i上 (Ebcdic)

半角文字1バイト、全角文字2バイト
全半角切替時にシフトコードが必要

Windows上 (Unicode)

半角・全角関係なしに2バイト
一部特殊な文字はそれ以上必要

Windows上 (ANSI)

※Delphi/400 V2007以前
半角文字1バイト、全角文字2バイト

→ [Unicodeについては次セッションにて追加情報があります。](#)

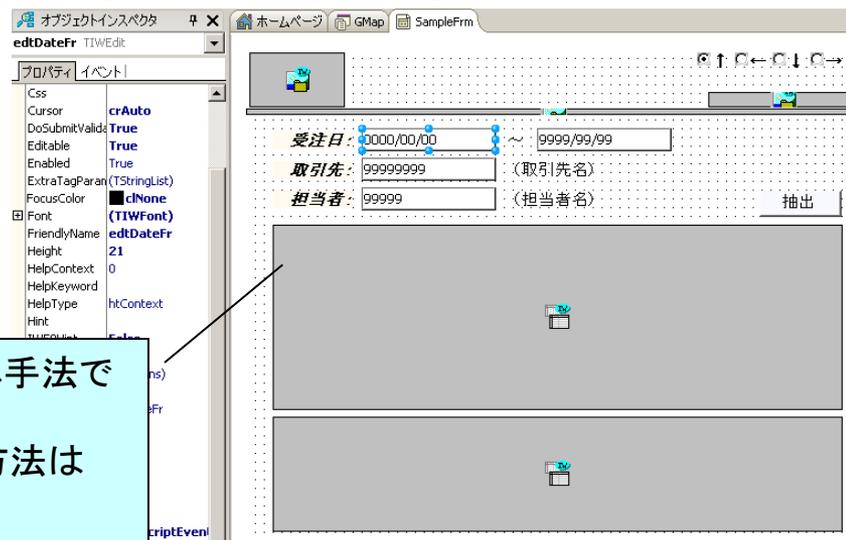
Webアプリ開発実演

■ 4) Delphi/400のWebアプリ開発

● VCL for the Web (IntraWeb)

VCL for the Web (IntraWeb)とは

Web画面の作成をC/Sプログラムを作るようなGUIの手法で作成できる機能です。通常のWebプログラムに必要なHTML、CSS、JavaScriptを知らなくても、Webアプリケーションを作成することができます。



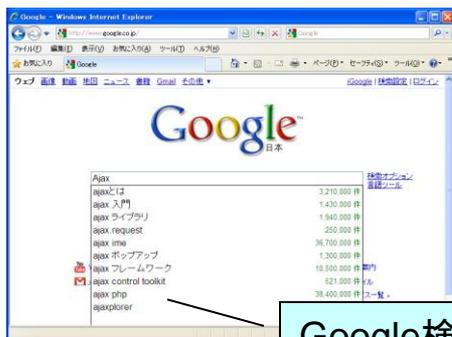
C/Sプログラムと同じ手法で
Web開発
プロジェクトの作成方法は
[補足資料①](#)を参照

■ 4) Delphi/400のWebアプリ開発

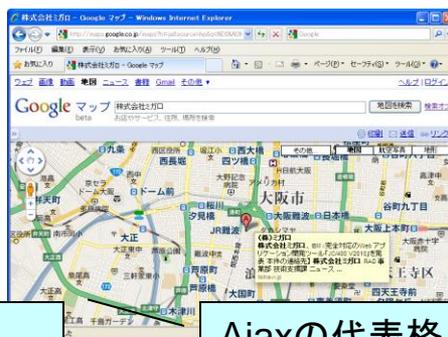
● VCL for the Web (IntraWeb)

VCL for the Webの新機能

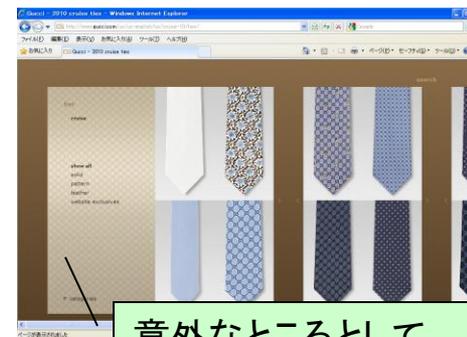
GoogleマップやGoogle検索の入力候補リストに代表されるWebの通信技術である**Ajax (Asynchronous JavaScript + XML)**をサポートしました。Ajaxは名前の由来から見てわかるようにJavaScriptで作成された技術ですが、VCL for the Webを使えば、**JavaScriptを1行も記述せずに**Ajaxの機能を利用できます。



Google検索の
検索キーワード表示



Ajaxの代表格
Google Map



意外なところとして
Gucciのオンラインショップ

4) Delphi/400のWebアプリ開発

● 対象となるWebアプリ

① 受注一覧照会

VCL for the WebによるWeb
作成します。C/Sアプリな

【実装機能】

- ・受注日項目を抜ける際の日
- ・取引先、担当者を抜ける際
- ・受注一覧のタイトルクリック
- ・受注一覧の受注番号クリッ

受注日: 2009/11/15 ~ 9999/99/99

取引先: 10000000 東京工業紳

担当者: 00001 山本 雅治

抽出

受注No	受注日	取引先	担当者	備考
100001	2009/11/15	東京工業紳	山本 雅治	納期前日にチェック
100003	2009/11/16	東京工業紳	山本 雅治	
100004	2009/11/17	東京工業紳	山本 雅治	
100005	2009/11/18	東京工業紳	山本 雅治	
100008	2009/11/18	東京工業紳	山本 雅治	
100009	2009/11/18	東京工業紳	山本 雅治	

行No	品名	数量	単価	金額
1	マウス(シリアル)	10	850	8,500
2	マウス(ワイアレス)	10	3,000	30,000
3	プリンター(ドット)	1	125,000	125,000

■ 4) Delphi/400のWebアプリ開発

● 対象となるWebアプリの目的

②取引先マスタ保守

取引先一覧

コード	名称	略称	郵便番号	住所1	住所2	住所3
10000000	東京工業㈱	東京工業㈱				
20000000	株式会社浜松機械	浜松機械				
30000000	大阪商事 株式会社	大阪商事㈱				
40000000	名古屋精密機器工業株式会社	名古屋精密				
50000000	株式会社浪速機器販売	浪速機器販売				

郵便番号検索

郵便番号: 5560017

5560005: 大阪府大阪市浪速区日本橋
5560006: 大阪府大阪市浪速区日本橋東
5560004: 大阪府大阪市浪速区日本橋西
5560017: 大阪府大阪市浪速区湊町
5560016: 大阪府大阪市浪速区元町

地図表示

都道府県: 大阪府
市区町村: 大阪市浪速区
町域等: 湊町2-1-57

データ更新

保存 取消

■ 5) Webアプリ開発① - Ajaxの利用

● Ajax対応前後の動作の違い

Ajax以前の動作

抽出ボタンを押下

各項目を入力

全画面を更新

Ajax以降の動作

項目を入力後、フォーカスを移動

必要箇所だけ更新

■ 5) Webアプリ開発① – Ajaxの利用

● 受注一覧照会の作成

受注一覧照会のポイント

<Ajaxの機能を使う>

→ **OnAsyncXXXX**イベントを利用するだけ！！



```
//受注日 (From) からフォーカスが抜ける際のAjaxを使ったイベント  
procedure TfrmWebSample.edtDateFrAsyncExit(Sender: TObject;  
    EventParams: TStringList);  
begin  
    // --- フォーカスを抜ける際に書式を整える  
    edtDateFr.Text := DateToDisplayString(edtDateFr.Text);  
end;
```

1/1

フォーカス移動

2009/01/01

■ 5) Webアプリ開発① – Ajaxの利用

● なぜ、Ajax？

Ajaxを使うメリット

Ajaxを使うメリットは以下のようなものが挙げられます。

■ 非同期通信を使うことによって**サーバ処理の待ち時間を短縮**

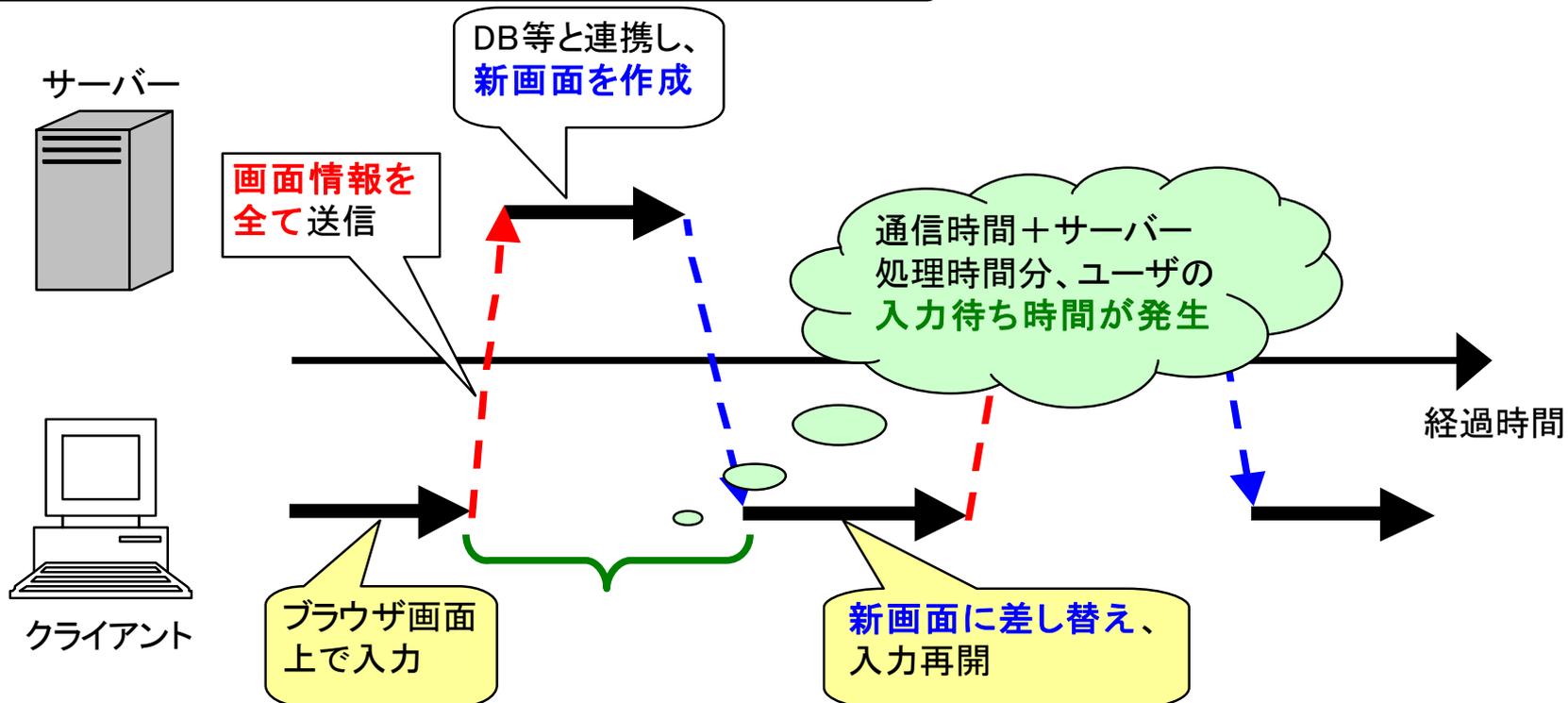
■ 必要箇所のデータのみをC/S間でやり取りすることで**通信負荷を軽減**

■ 処理の一部をクライアント端末に代替させることで**サーバ負荷を軽減**

■ 5) Webアプリ開発① - Ajaxの利用

● Ajaxのメリット

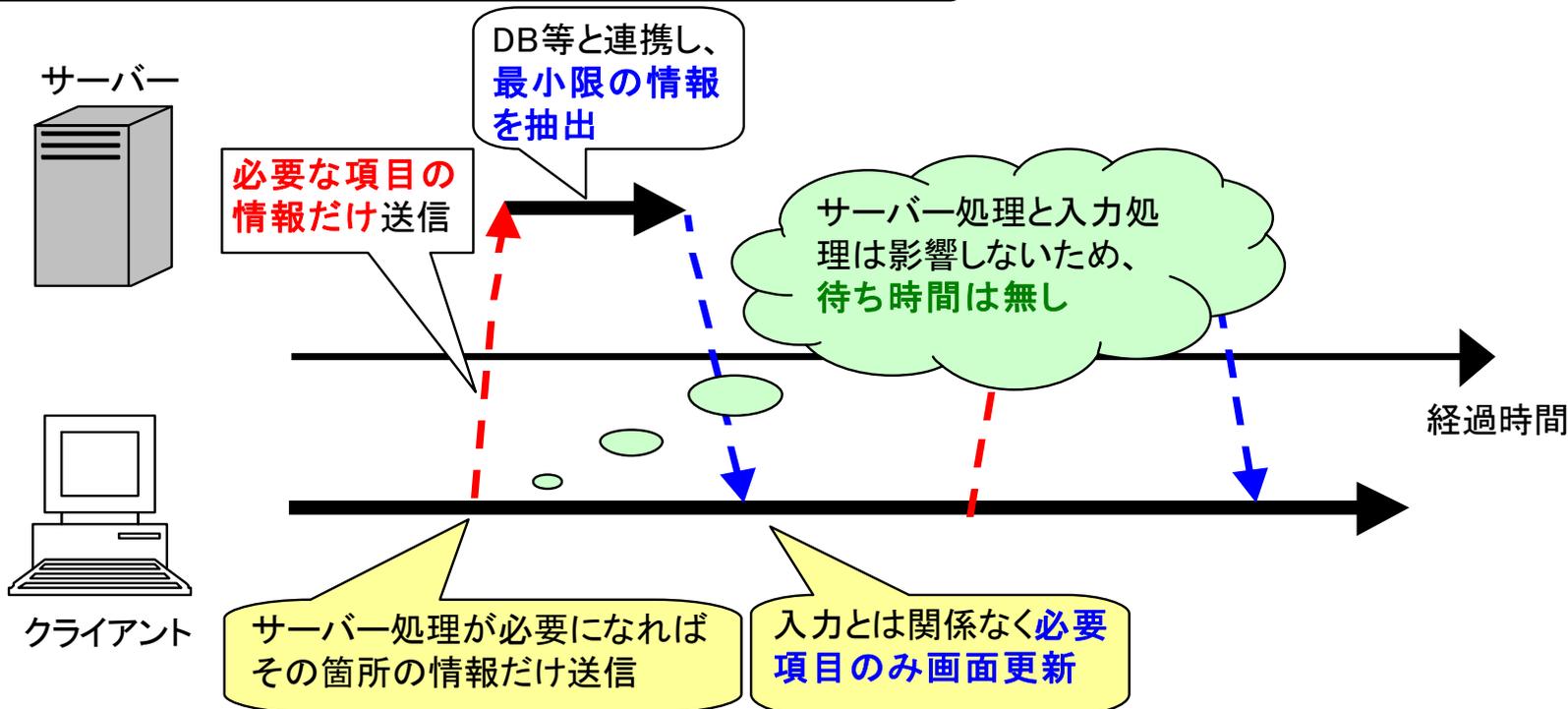
従来よく利用されていた同期通信のイメージ



■ 5) Webアプリ開発① - Ajaxの利用

● Ajaxのメリット

Ajaxでよく利用される非同期通信のイメージ



■ 6) Webアプリ開発② – Webサービスの利用

● どのようなWebサービスが存在するか

Google Web APIs

検索や地図表示、スケジュール機能等、様々なサービスを提供しています。

Yahoo!デベロッパーネットワーク

検索は勿論、テキスト解析やオークション・ショッピング等のサービスを提供しています。

Amazon Web サービス

Amazonで取り扱っている商品の検索を行えます。

他にも郵便番号検索や為替レート取得、PDF変換/作成等もあります。

■ 6) Webアプリ開発② – Webサービスの利用

● 今回利用するWebサービスについて

郵便番号検索

今回利用する郵便番号検索サービスは3桁以上の郵便番号を渡すと前方一致検索で対象となる全ての郵便番号を返してくれるのが特徴です。

参考: <http://groovetechnology.co.jp/web/service/zipsearch/index.html>

一般的なWebサービスと同じくHTTPを使って検索したい郵便番号を渡すとXMLの形でデータを返してくれますので、Delphi/400付属のXMLデータバインディングウィザード等を併用しつつ、データを利用します。

参考: 補足資料②

```
- <address>
  <zipcode>5560017</zipcode>
  <prefecture>大阪府</prefecture>
  <city>大阪市浪速区</city>
  <town>湊町</town>
  <prefecture_yomi>オオサカフ</prefecture_yomi>
  <city_yomi>オオサカシナニワク</city_yomi>
  <town_yomi>ミナトマチ</town_yomi>
</address>
```

XMLとして取得されるデータ

検索結果として取り込み!



■ 6) Webアプリ開発② – Webサービスの利用

● 今回利用するWebサービスについて

地図表示

Google Maps APIにて生成される地図表示を利用します。
マウスのドラッグを使った地図表示やマーカー設定等多彩な機能を持つ地図サービスです。

こちらは画面上へ地図表示する専用のJavaScriptを付加する必要がありますので、[WebBroker](#)を使って作成したCGIで表示用のHTMLを作成し、VCL for the WebのIWURLWindowコンポーネントを使って表示します。

参考: [補足資料③](#)

IWURLWindowコンポーネント
で読み込んだHtmlを生成



■ 6) Webアプリ開発② – Webサービスの利用

● 取引先マスタ保守の作成

取引先マスタ保守のポイント

<郵便番号検索を使う>

```
//指定された郵便番号をもとにWebサービスから住所を取得する
procedure TfrmWebSample2Input.GetAddrList(sZip: String; bListup: Boolean);
var
  i: Integer;
  URI: String;
  RootNode: IXMLGroovewebstype;
begin
  // *** グループテクノロジー株式会社様の郵便番号検索APIを利用します
  //      グループテクノロジー Webサービス
  //      URL: http://groovetechnology.co.jp/webservice/

  // --- Webサービスを利用して住所の一覧を取得する
  URI := 'http://groovetechnology.co.jp/ZipSearchService/v1/zipsearch?'
        + 'zipcode=' + sZip + '&format=xml&ie=Shift_JIS&oe=Shift_JIS';
  // レスポンスとして戻されたXMLを処理する
  xdc.XML.Text := ihpZIP.Get(URI);
  RootNode := Getgroovewebstype(xdc);
```

■ 6) Webアプリ開発② – Webサービスの利用

● 取引先マスタ保守の作成

取引先マスタ保守のポイント

<地図表示を使う>

```
//地図表示ボタンがクリックされた際の通常イベント
procedure TfrmWebSample2Input.btnMapClick(Sender: TObject);
begin
    // *** CGIプログラム連動 ***
    // --- 別途作成したGoogle Maps APIを利用するためのHTMLを生成する
    //     CGIプログラムをURLWindowに表示する
    wndMap.URI := cCGIURL + '?addr='
        + HTTPEncode(edtMCADR1.Text + edtMCADR2.Text + edtMCADR3.Text);
    wndMap.Visible := True;
end;
```

Google Maps APIを利用する
htmlを返すCGIのURL

補足資料

VCL for the Webで新規プロジェクトを作成する

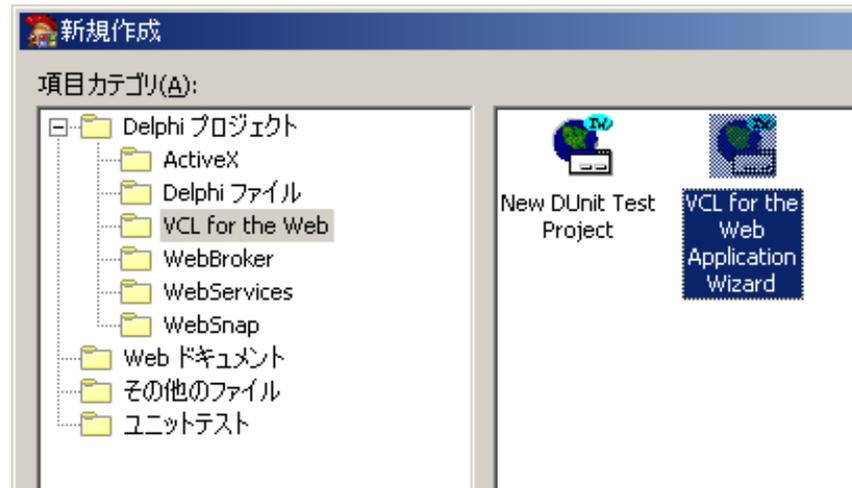
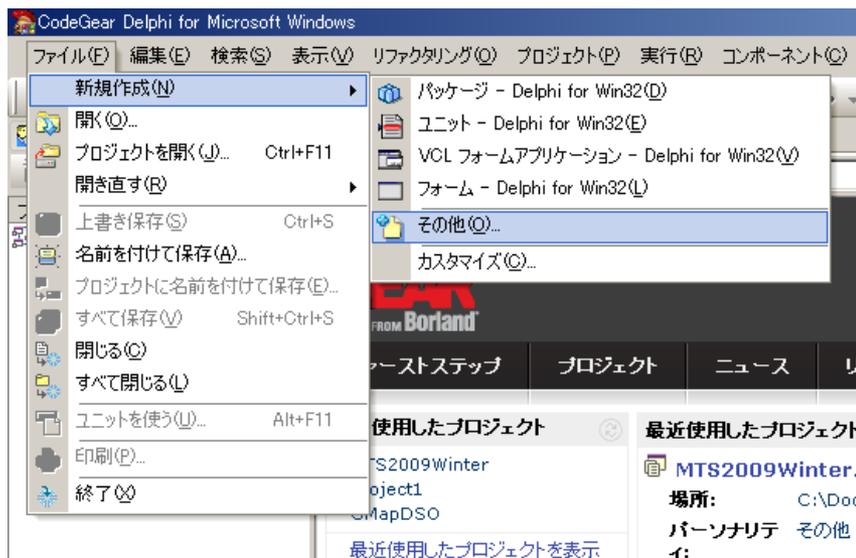
データバインディングウィザードを利用する

WebBrokerで新規プロジェクトを作成する

■ 補足資料

● VCL for the Webで新規プロジェクトを作成する①

- ✓ メインメニューから[ファイル | 新規作成 | その他]を選択
- ✓ 項目カテゴリから[Delphiプロジェクト | VCL for the Web]を選択後、「VCL for the Web Application Wizard」を選択



■ 補足資料

● VCL for the Webで新規プロジェクトを作成する②

- ✓ 「VCL for the Web Application Wizard」で必要事項を選択・入力してOKを押下するとプロジェクトが生成されます。

The screenshot shows the 'VCL for the Web Application Wizard' dialog box. It has a title bar 'VCL for the Web Application' and a logo 'intraWeb The V'. The main text says 'Welcome to the VCL for the Web Application Wizard. Select the application you want to create and any additional options'. There are two main sections: 'Application Type' and 'Options'. In 'Application Type', 'StandAlone Application' is selected with a radio button. In 'Options', 'Create User Session' is checked, and 'Use ServerManager Lite' and 'Use ISAPIThreadPool' are unchecked. The 'Main Form' dropdown is set to 'Standard 4.0'. At the bottom, 'Project Name' is 'Project1' and 'Project Directory' is 'xxxxxx\My Documents\RAD Studio\Projects\Project1\'. There are 'Ok' and 'Cancel' buttons at the bottom right.

通常、「Create User Session」のチェックはつけたままにする

デバッグ実行用モジュール
StandAlone Application

IIS用モジュール
ISAPI Extension

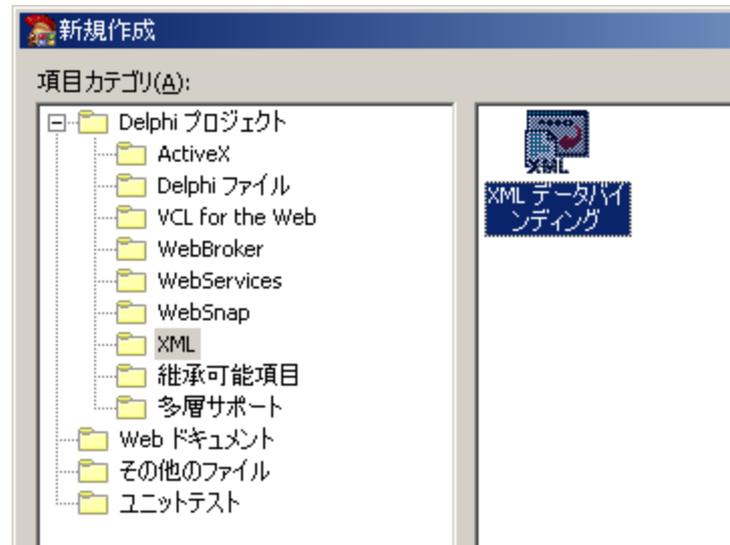
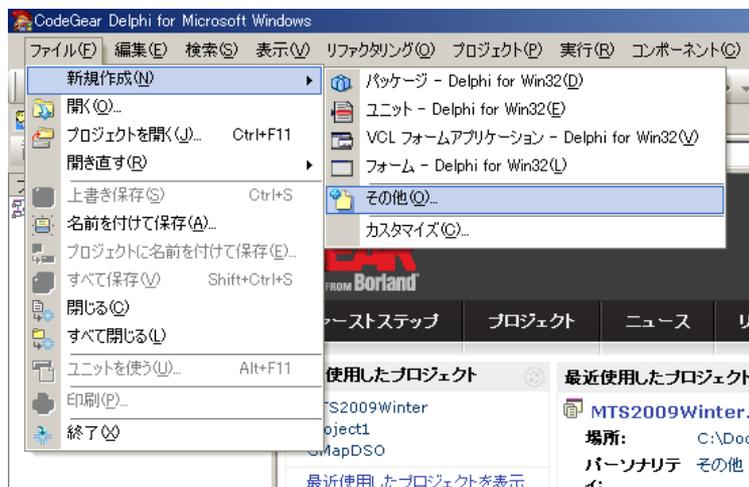
プロジェクト名や保存先を指定する

■ 補足資料

● データバインディングウィザードを利用する①

- ✓ メインメニューから[ファイル | 新規作成 | その他]を選択
- ✓ 項目カテゴリから[Delphiプロジェクト | XML]を選択後、「XMLデータバインディング」を選択

※項目カテゴリのXMLはプロジェクトが開かれた状態でないと表示されません。



■ 補足資料

● データバインディングウィザードを利用する②

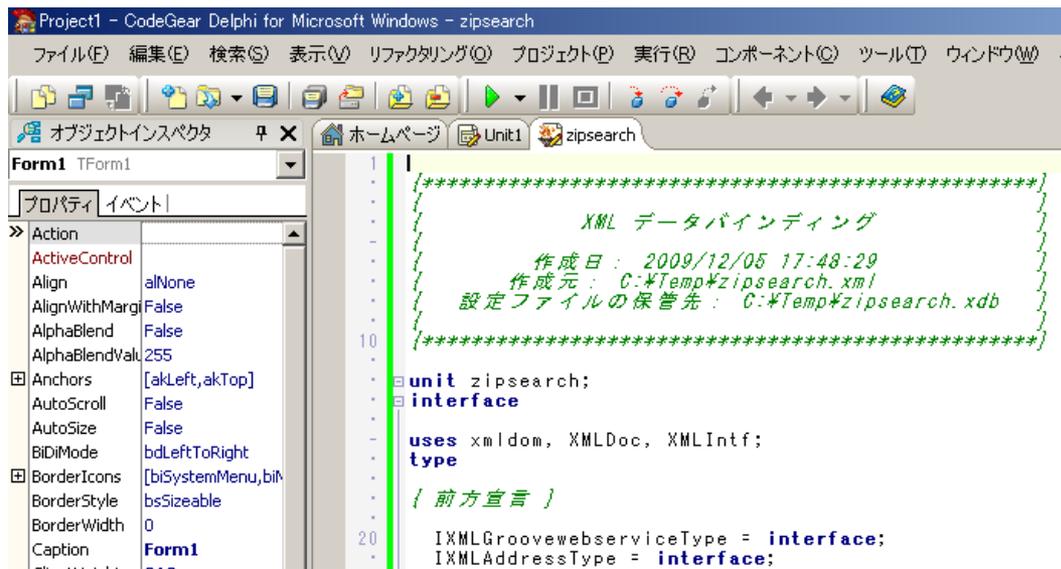
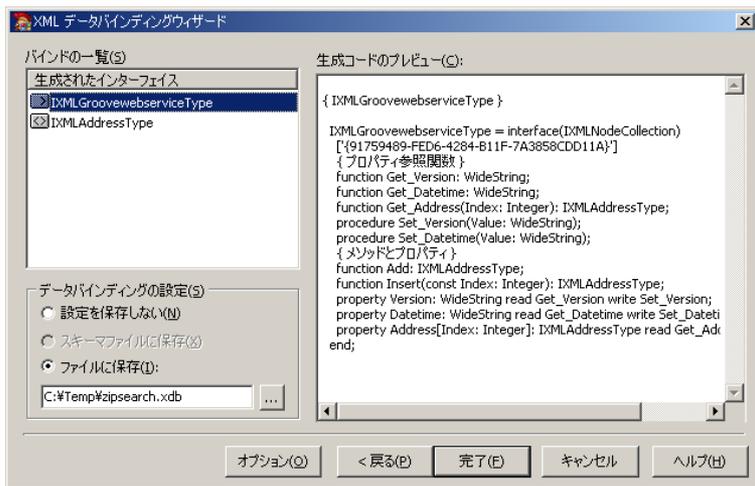
- ✓ データバインディングウィザードでスキーマファイルまたはXMLファイルを選択
- ✓ 解析結果が表示されるので、必要に応じて修正を加えて次へ



■ 補足資料

● データバインディングウィザードを利用する②

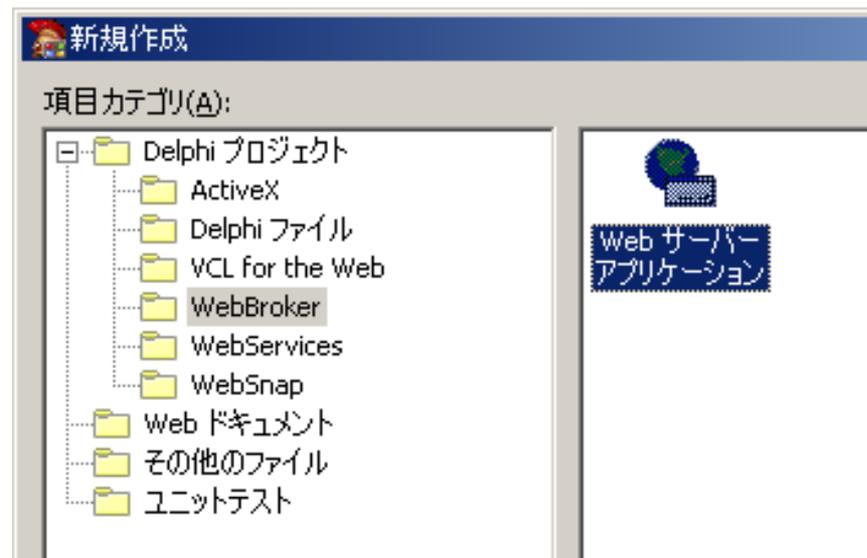
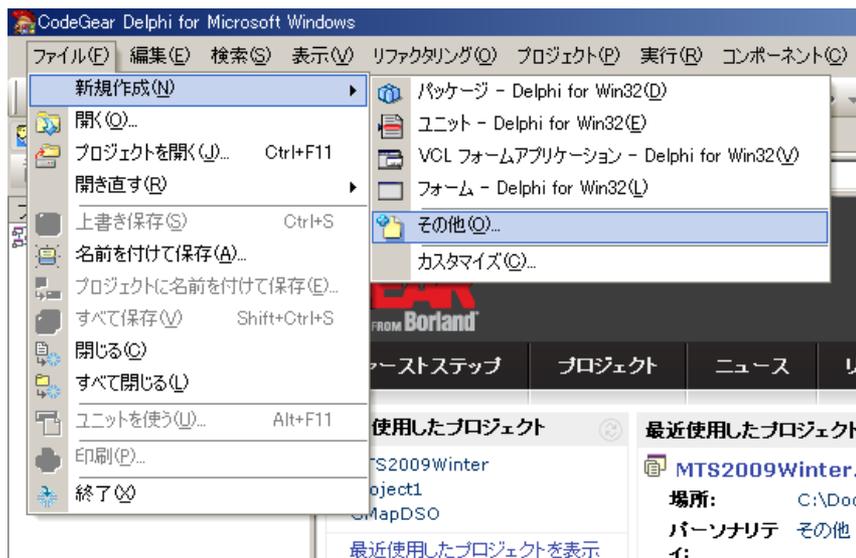
- ✓ 解析結果に従いコードがプレビューされるので、完了を押下するとソースコードが生成されます。



■ 補足資料

● WebBrokerで新規プロジェクトを作成する①

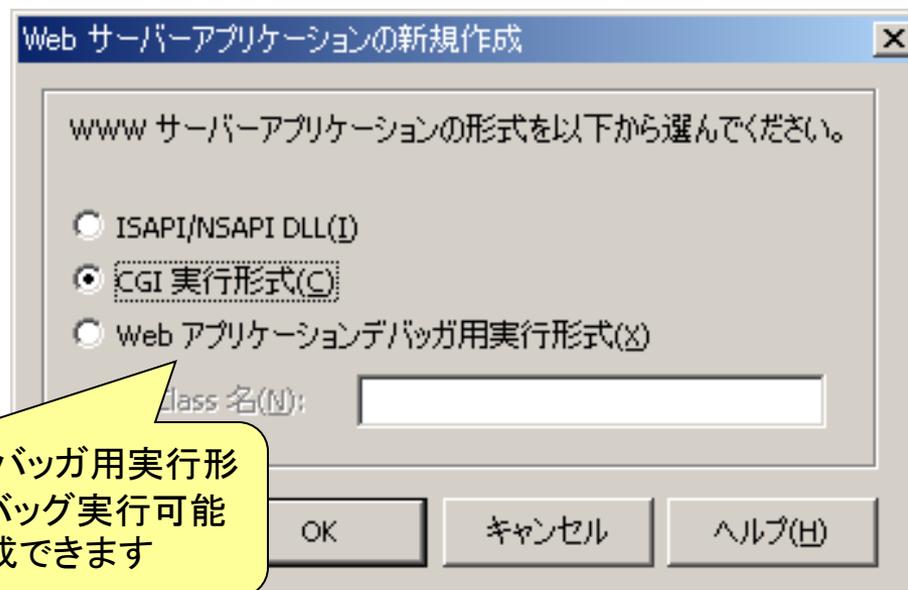
- ✓ メインメニューから[ファイル | 新規作成 | その他]を選択
- ✓ 項目カテゴリから[Delphiプロジェクト | WebBroker]を選択後、「Webサーバーアプリケーション」を選択



■ 補足資料

● WebBrokerで新規プロジェクトを作成する②

- ✓ 「Webサーバーアプリケーションの新規作成」で作成したいWebサーバーアプリケーションの形式を選択してOKを押下するとプロジェクトが生成されます。



Webアプリケーションデバッガ用実行形式を選択することでデバッグ実行可能なモジュールを作成できます