【B4】Delphi Prismチュートリアルセッション



### Delphi Prism による Blackfish SQL を利用した Webサービス/ASP.NETアプリケーションの作成

株式会社ドリームハイブ 代表取締役 山本 悟

スピーカープロフィール



#### 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、最後のスライドでご紹介する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。

http://www.dreamhive.co.jp/ http://dhive.jp/blog/yama/ http://mixi.jp/show\_friend.pl?id=104357

アジェンダ



- このセッションの成果物と前提知識
- 早速作りましょう
  - WebサービスとWebアプリケーション
- DBに繋げる場合
  - Blackfish SQLを使います
- 終わりに

- このセッションは、テクニカル・チュートリアルセッションです
- Delphi PrismとBlackfish SQLを利用した、WebサービスとWebアプリケーションを実装します
- 開発環境から運用環境に移行する際のポイントを紹介します
- 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



このセッションの成果物と前提知識



- データベースの結果を返すWebサービス
- Webサービスに接続して、画面を表示するWebアプリケーション



前提知識



• Webサービス

XML Web サービスとは、XML、HTTP、SOAP などのインターネット標準技術を利用して、異なるプラットフォーム上のアプリケーションとも統合することが可能なソフトウェアの総称です(MSDNから)





#### 早速作りましょう



- 1. まずは簡単なWebサービスから
- 2. Webサービスに接続するWebアプリケーション





- まずは簡単なWebサービスから
  - [ファイル|新規作成|Web サイト]から「<u>ASP.NET Webサービス</u>」を選択、
     言語はもちろん「<u>Oxygene</u>」



- Methodのimplementation部には「[WebMethod]」を忘れずに
- App\_Codeのpasファイル名やクラス名を変更するときは、 asmxファイルのWebServiceディレクティブのCodeBehind属性とClass 属性を合わせること

# 早速作りましょう - その1のソース



/Service.asmx
<%@ WebService Language="Oxygene" CodeBehind="~/App\_Code/Service.pas" Class="Service" %>
/App\_Code/Service.pas
type
[WebService(&Namespace := 'http://tempuri.org/')]
Service = public class(System.Web.Services.WebService)
public
 method HelloWorld: string;
 method GetSum(a, b: Integer): Integer;
end;
implementation

```
[WebMethod]
method Service.HelloWorld: string;
begin
  result := 'Hello World';
end;
```

#### [WebMethod]

```
method Service.GetSum(a, b: Integer): Integer;
begin
  result := a + b;
end;
```



- Webサービスに接続するWebアプリケーション
  - ・ [ファイル|追加|新しいWebサイト]から「ASP.NET Web サイト」を選択
  - [Web サイト|Web 参照の追加]を選択、プロジェクトにWeb 参照を追加

Web 参照の追加		2 ×
Web サービス URL に移動して [参照の追加] をクリックし、使用可能なサービスをす^	て追	加します。
③ 戻る ● ■ ₫ 益		
URL(U): http://localhost:65261/WebSite8/TestWebService.asmx	移	助
TestWebService	Î	この URL で見つかった Web サービス( <u>S</u> ): 1 個のサービスが見つかりました: ^
以下の操作はサポートされています。正式な定義に関しては、 <u>サービスの説明</u> を参照し てください。		- TestWebService
GetSum     HelloWorld	ш	
この Web サービスは、http://tempuri.org/ を既定の名前空間として使用し ています。		↓ Web 参照名( <u>N</u> ):
推奨:XML Web サービスをパブリックにする前に、既定の名前空間を変更してく ださい。		localhost
各 XML Web サービスは、クライアント アプリケーションが Web 上の別のサービスから 聞別できるように、一部の名前空間を必要します。http://tempuri.org/1ま,開発中の XML Web サービスで使用できますが、発行されている XML Web サービスには、より永 久的な名前空間を使用する必要があります。		参照の追加(民)
XML Web サービスを管理する名前空間によって講訳しなければなりません。たとえば、 会社のインターネットドメイン名を名前空間の一部として使うことができます。XML Web サービス名前空間の多くは、URL の形式に変化していますが、Web との実際のリソース にポイントする必要はありません。(XML Web サービス名前空間は URI です)		
ASP.NETを使用する XML Web サービス作成用では、WebService 層性の Namespace プロ・行べを使って現在の名前空間を実更できます。WebService 間当は、 XML WebP レビス メンバを含むクラスに適用れる層性です。以下は、名前空間を "http://microsoft.com/webservices/" に設定する、コード サンブルです。	+	
< )		キャンセル

- TextBoxを2つ、Buttonを1つフォームに配置
- ButtonのClickイベントに、GetSum()を利用したコードを記述

早速作りましょう - その2のソース



/Default.aspx.pas

method \_Default.Button1\_Click(sender: System.Object; e: System.EventArgs);

var

ws: localhost. Service;

begin

ws := new localhost. Service;

Button1.Text := ws.GetSum(Integer.Parse(TextBox1.Text), Integer.Parse(TextBox2.Text)).ToString; end;





- 1. 運用環境にOxygeneは入っているか?
  - VB.NETやC#は、.NET Frameworkのインストール時に利用できるよう に構成されているが、Oxygene(Delphi Prism)は構成されていない
    - /Default.aspx のディレクティブ

       <%@ Page Language="Oxygene" AutoEventWireup="true"
      CodeFile="Default.aspx.pas" Inherits="\_Default" %>
- 2. WebサービスはUDDIが前提になっているから、サーバの特定 ができないのでは?
  - Web参照で追加された、wsdl / disco ファイルのどちらにもWebサービスへのURLが直接記述されている





## 1. プリコンパイルしておくのが簡単

 [ビルド|Web サイトの発行]を選択し、「このプリコンパイルされたサイト を更新可能にする」チェックボックスのチェックを外す

Web サイトの発行
ターゲットの場所( <u>L)</u> : (ftp://、http:// または ドライブ:¥パス)
kryu2¥My Documents¥Visual Studio 2008¥Projects¥WebServiceTest¥PrecompiledWeb¥WebSite10
□ このブリコンパイルされたサイトを更新可能にする(合)
固定名およびシングル ページ アセンブリを使用する(E)
デバッグ情報の作成( <u>G</u> )
プリコンパイル済みアセンブリで厳密な名前を有効にする(S)
◎ 厳密な名前ツールで生成されたキー ファイルを使用する(№)
キー ファイルの場所:
○ キー コンテナを使用する(I)
キーコンテナ:
□ アセンブリを AllowPartiallyTrustedCallerAttribute (APTCA) に設定する( <u>M</u> )
ок <i><b>‡</b>т&gt;tzıl</i>

• 運用環境にOxygeneをインストールする方法もある





#### Wsdl.exeを利用する Webサービス接続用のプロクシクラスをWsdl.exeで作成し、サーバ情報はWeb.config ファイルに外出ししてしまう

1.「ソリューション エクスプローラ」でクラスファイルを作りたいプロジェクトを選択し、右ク リックメニューの「ASP.NET フォルダの追加」から、「App\_Code」を選択する 2.「Visual Studio 2008 コマンドプロンプト」を立ち上げ、手順1の App\_Code フォルダに 移動する 3.「Wsdl.exe /urlkey:<構成キー名> /!:<言語名> <URL>」を入力、実行

<構成キー名>:Web.config の appSettings セクションで指定したい名前 <言語名>:クラスファイルを記述する言語名(C# や Oxygene など) <URL>:Web サービス提供の URL (~.asmx など)

例) wsdl.exe /urlkey:ws /l:Oxygene <u>http://localhost:55555/<WebSite>/Service.asmx</u> 4.Web.config内の「<configuration>¥<appSettings>」に「構成キー」を追加 例)<add key="ws" value="http://localhost:55555/<WebSite>/Service.asmx"/>

### 解決方法 – ソース







#### **DEVELOPER CAMP**

DBに繋げる場合



DBのデータを返すWebサービスを作る
 DBのデータを受け取るWebアプリケーションを作る





- DBのデータを返すWebサービスを作る
  - 1. DBファイルを「/App\_Data」に配置する(作成する)
    - 1. 「ツール|データベースへの接続」を選択し、「データソース」を「Blackfish SQL」に、「データプロバイダ」をBlackfish SQL Remote Data Provider for .NET」に設定し、「続行」する
    - 接続情報を入力し、「OK」をクリックする 新規に作成する場合は、「詳細設定」で「Create=True」にする

例)host=LocalHost;user=sysdba;password=masterkey;database="Path ¥To¥App\_Data¥TestData"

※「SQLウィンドウ」もしくは「クエリ」で、SQLのチェックなどができる

3. Web.config に<connectionString>を追加する

### DBに繋げる場合 – その1 – 画面





OK

キャンセル

#### ? X 選択されたデータ ソースに接続するための情報を入力するか、または [変更] をクリックして、 別のデータ ソースおよび (または) プロバイダを選択します。 Blackfish SQL (Blackfish SQL Remote) 変更(<u>C</u>)... LocalHost ........ データベース: 2008¥WebSites¥WebService1¥App\_Data¥TestData.jds 参照(B) 詳細設定(V)... OK キャンセル

新規にDBを作る場合に必要な設定





 [表示|プロパティページ]を開き、「参照設定」で 「Borland.Data.BlackfishSQL.RemoteClient」を追加する

in juzzi, bocument					-	
用197年	A10.7	<b>P</b> 1	d-state			
- 15 - ザー補助	Borland.Data.BlackfishSQL.RemoteCli	ent GAC	8.0.0.0			
8オプション iBuild オプション						
			(D)	a art	9/p)	38.95/11)
			3ELOU(L2)		»( <u>D</u> )	3E87(U)
			E	ОК	キャンセル	油用(6)

- 3. /App\_Code/Service.pasのusesにデータアクセス用の名前空間を足す
  - System.Data,
  - System.Data.Common,
  - System.Data.SqlClient,
  - Borland.Data.DataStore
- 4. Web.configのconnectionStringを取得するconstructorを追加
- 5. /App\_Code/Service.pasにDataTable型を戻り値としたGetTableメソッドを追加

### DBに繋げる場合 - その1



■DBスキーマ	
CREATE TABLE TEST(	∎/Web.config
ID INTEGER AUTOINCREMENT PRIMARY KEY,	<connectionstrings></connectionstrings>
TEXTCLM VARCHAR(16),	<add <="" name="ws" td=""></add>
INTCLM INTEGER,	connectionString="host=localhost;database= DataDirectory ¥TestData.jds;use
DATETIMECLM TIMESTAMP	r=sysdba;password=masterkey nervider#lama="Parland Data PlackfickSOL ParasteClient"/>
);	providentanie= Bonand.Data.BiackishSQL.Remotechent />
INSERTINTO TEST (TEXTCLM, INTCLM, DATETIMECLM) VALUES (Text1,	-lànn CodelSenice nac
100, <u>DATE'2009-7-2'</u> );	aublic
INSERTINTO TEST"(TEXTCLM", "INTCLM", "DATETIMECLM") VALUES(Text2,	publik. Assertionation
102, DATE 2009-7-14 ); THEFT THE EXCEPTION (TTUTE) (TT	constructor,
INSERTINTO TEST (TEXTCLM, INTCLM, DATETIMECLM) VALUES(Text3, 200, DATE 2009-7-22);	conchrister Service
INCERTINED TECT"/ TEVTO MILITATO MILITATE TIMEO MILITA (TEVHA)	berin
102. DATE 2009-8-1'):	celf ConnectionString :=
INSERT INTO TEST"("TEXTCLM", "INTCLM", "DATETIMECLM") VALUES('Text5',	System.Configuration.ConfigurationManager.ConnectionStrings[ws].Connectio
200, DATE 2009-8-30');	nString;
	end;
	[WebMethod]
	method Service.GetTable: DataTable:
	heain
	using dsda: DataStoreDataAdapter := new DataStoreDataAdapter(`SELECT *
	FROM TEST', self.ConnectionString) do begin
	using dt: DataTable := new DataTable(TestTable) do begin
	dsda.Fill(dt);
	result := dt;
	end;
	end;
	end;





- DBのデータを受け取るWebアプリケーションを作る
  - 1. プロキシクラスをWsdl.exeによって更新する(再作成する)
  - 2. フォームにGridViewを1つ配置する
  - 3. GridViewにWebサービスから返されるDataTableをDataSourceプロパ ティに代入し、DataBindメソッドを呼び出すコードを、Page\_Loadイベン トなどに記述する

DBに繋げる場合 - その2 - ソース



Default.aspx.pas
method \_Default.Page\_Load(sender: Object; e: EventArgs);
var
ws: Service;
begin
ws := new Service;
GridView1.DataSource := ws.GetTable;
GridView1.DataBind;
end;





- Blackfish SQLには接続(コネクション)数の制限がありませんで したっけ?
  - Professional版は、1ユーザー/4コネクション、512 MBデータベースサイズをサポートするBlackfish SQLデスクトップデータベースの無制限配布ライセンス
  - Enterprise / Architect版は、5ユーザー/20コネクション、2 GBデータ ベースサイズをサポートするBlackfish SQLデータベースの無制限配布ラ イセンス
- 2. 問題が起きているのはDB?どうやってチェックする?
- 3. 配布は簡単?





### 1. きちんと毎回接続を切りましょう

複数のクライアントから接続される可能性が高いのであれば「リモート接続」を利用し、毎回コネクションを切りましょう

→Blackfish SQLはサービスで起動する必要がある

#### 例)毎回コネクションを切るコード

begin

```
for n: Integer := 0 to 30 do begin
    using conn: DataStoreConnection := new
    DataStoreConnection('host=localhost;database=path¥to¥TestData.jds;user=sysdba;password=mast
    erkey" providerName="Borland.Data.BlackfishSQL.RemoteClient') do begin
    conn.Open();
    conn.Close();
    end;
end;
```

end;

### 解決方法



- 2. Blackfish SQLにもログ(システム ログ)機能があります
  - C:¥Program Files¥CodeGear¥Delphi Prism¥Bin¥BSQLServer.exe.config ファイル内の次の値を設 定します
    - <configuration>¥<appSettings>¥<add key=blackfishsql.logFile" value="ログファイルのパス">
    - <configuration>¥<appSettings>¥<add key=blackfishsql.logFilters" value="ログのフィルタ文字列">
      - フィルタ文字列
         INIT,CONNECT,PREPARE,SET\_PARAMETER,EXECUTE,TRANSACTIONS,ERRORS,LOCK\_ERRORS,SY
         STEM
    - ※Blackfish SQLがサービスで起動している場合は、サービスを再起動させる必要があります

#### 3. 本番環境への配布方法

- 配布するファイル
  - BSQLServer.exe
  - BSQLServer.exe.config(設定ファイル:必要があればコピーして編集、設定する)
  - Borland.Data.BlackfishSQL.RemoteClient.dll
  - Delphi Prism BlackfishSQL.slip
  - ・ /App\_Data内のデータベースファイル(本体、ログなど)
- DBプロバイダへのアセンブリ登録(DbProviderFactories)
  - Web アプリケーションごとに設定するなら「Web.config」ファイルへ登録
  - Web サーバで一括して設定する設定なら、「Machine.config」ファイルへ登録
  - C:¥WINDOWS¥Microsoft.NET¥Framework¥<バージョン>¥CONFIG¥Machine.config
  - <configuration>¥<system.data>¥<DbProviderFactories>
- SCコマンドを使えば、配布時のサービス設定を自動化できます





#### ■コマンド例

sc failure "BlackfishSQL" reset= 900 actions= restart/180000/restart/180000/restart/180000

(ローカル コンピュータ) Bl	ackfishSQI	Lのプロパティ	? 🔀	(ローカル コンピュータ) B	lackfishSQ	Lのプロパティ	?
全般 ログオン 回復 依存 このサービスがエラーになった場合	『関係』 合のコンピュータ	の応答を指定してください。		全般 ログオン 回1 ( 依不 このサービスがエラーになった場	子関係 合のコンピュータ	の応答を指定してください。	
最初のエラー( <u>F</u> ):	-(E): 何もしない			最初のエラー( <u>F</u> ):	サービス	サービスを再起動する	
次のエラー( <u>S</u> ):	何もしな	()	~	次のエラー( <u>S</u> ):	サービスを再起動する		~
その後のエラー(山):	凹: 何もしない			その後のエラー( <u>U</u> ):	サービスを再起動する		<b>~</b>
エラーカウントのリセット(Q): サービスの再起動(V):	0	日後に行う       分後に行う		エラー カウントのリセット(Q): サービスの再起動(V):	0 3	日後に行う 分後に行う	
- プログラムの実行 プログラム( <u>P</u> ):		参照(8)		- プログラムの実行 プログラム( <u>P</u> ):		参照但	)
コマンド ラインのパラメータ(2)	» [			コマンド ラインのパラメータ(	2		
של - פּבזטערא אעאנ	ウントのオプショ	ンを追加 (/fail=%1%( <u>E</u> )		ל-לבושעאל אעצים	コウントのオプショ	」ンを追加 (/fail=%1%)( <u>E</u> )	
		コンピュータの再起動のオブション(R)				コンピュータの再起動のオブション(	<u>R</u> )
	ОК	キャンセル 適	5月( <u>A</u> )	L	ОК	++>\UM	適用( <u>A</u> )

それと



- App\_Dataフォルダのセキュリティ設定
  - Blackfish SQLデータベースのファイルを保存する「App\_Data」のフォルダ・ ファイルのセキュリティ設定には、ASP.NETの動作アカウントを追加する必要 があります
  - ASP.NETの動作アカウントはOSによって違いますが、Windows XP では 「ASP.NET」、Windows Server 2003 では「NETWORK SERVICE」ユーザと なっています
  - ASP.NETの動作アカウントへのアクセス許可には、「変更」を追加しましょう
- 名前空間の設定
  - /App\_Code/Service.pas内の [WebService(&Namespace := '<u>http://tempuri.org/</u>')]は、 自社のURL等を指定しましょう ※tempuriのままだと警告メッセージが出ます



# 終わりに



- ご協力いただいた方
  - 株式会社クリエ・イルミネート 溝端氏 <u>http://www.crie-illuminate.jp/</u>
    - WCFやUDDIなどの情報を教えていただきました
  - DEKO(深夜の誤爆)氏
    - Blackfish SQLの配布についてなどの情報を教えていただきました





事業内容概要図

「日本一相談しやすい中小企業向けITコンサルティング」を目指す「システム開発」会社です。

商号 株式会社ドリームハイブ
設立 2001年12月4日
3つの事業を組み合わせることにより、
中小企業が抱えるIT関連の負荷を軽減し、
仕事を楽にするサービスを提供しています

 「コンサル
ティング
トレーニング

このスライドを見たらアクセスしてみてください → <u>http://www.dreamhive.co.jp/</u>

公開アプリの紹介



#### <u>アプリ</u>

#### • DH-APPLI

- M シンプルなメモツール
- · Change! -お手軽名前変更ツール
- · GrepReplace -複数行一括置換
- · PlacesBar ダイアログを便利に!
- ColorPicker 色情報取得ツール
- ・ その他
  - ・ VALUE-DOMAIN 専用 DNS更新ツール



累計70,000件以上のダウンロード!



#### <u>アプリの配布先</u>

- 自社Web <u>http://dhive.jp/dhappli/</u>
- Vector <u>http://www.vector.co.jp/vpack/browse/per</u> <u>son/an037127.html</u>
- ・ 窓の杜 http://www.forest.impress.co.jp/lib/sys/file /filesearch/grprplc.html



# ありがとうございました