

Delphi OplossingsCourant

Vol. 9. No. 1, Een gratis elektronische publicatie van **Bob Swart Training & Consultancy (eBob42)** - <http://www.eBob42.com>



Helmond, 6 september 2007,

Welkom bij het nieuwe nummer van de Delphi OplossingsCourant. Precies een jaar na het vorige nummer. Toen werden de Turbo edities van Delphi, Delphi for .NET, C++Builder en C#Builder uitgebracht, en gisteren was het wederom tijd voor iets nieuws, namelijk CodeGear RAD Studio 2007 met daarin Delphi for Win32, C++Builder en Delphi for .NET 2.0/3.0. Dit is de opvolger van de Borland Developer Studio 2006 (voor Win32 en .NET 1.1). Let op: er komt geen aparte Delphi for .NET 2.0 uit, wie met ASP.NET 2.0 bezig wil zal echt de hele CodeGear RAD Studio 2007 moeten aanschaffen!

Dat kan nu overgens ook bij mij, want sinds deze zomer ben ik een officiële CodeGear Business Partner en Reseller voor de BeNeLux van de Electronic Software Delivery (ESD) edities van de CodeGear producten. ESD wil zeggen dat je – na betaling – een download link krijgt voor een setup.exe plus een serienummer om daarmee de betreffende versie van de ontwikkelomgeving te installeren. Dat doe ik voor Delphi 2007 for Win32, C++Builder 2007, Delphi for PHP en nu ook voor CodeGear RAD Studio 2007. Wie interesse heeft in andere CodeGear producten kan het beste even contact opnemen per e-mail voor meer informatie.

Wie een CodeGear product aanschafft heeft ook de mogelijkheid tot het gelijktijdig aangaan van een "Subscription" voor een periode van 12 maanden (vlak voor het eind van deze periode kun je dit steeds weer verlengen met weer 12 maanden). Waarom je dit zou willen, en wat je er voor krijgt, beschrijf ik verderop in dit nummer van de Delphi OplossingsCourant.

In dit nummer van de Delphi OplossingsCourant ga ik ook vast in op enkele nieuwe features van de CodeGear RAD Studio 2007, waaronder Generics.

Daarnaast laat ik ook Delphi 2007 for Win32 aan de orde komen, alsmede DBX4 – de nieuwste versie van de dbExpress data access laag die nu voor zowel Win32 als .NET het aangewezen framework is (en de Borland Data Provider for .NET zal vervangen). Ook enkele ervaringen met Delphi en Windows Vista deel ik graag in dit nummer.

SDC, EKon, EuroDevCon 2007

Wie meer wil zien van CodeGear RAD Studio 2007, die lang onder de naam Delphi Highlander in beta was, kan o.a. bij de Software Developer Conference op 17 en 18 september zijn hart ophalen.

Ook op de EKon 11 en CodeGear EuroDevCon in Frankfurt, van 24 t/m 28 september zal CodeGear RAD Studio te zien zijn. Op beide events zal ik ook sessies doen met CodeGear RAD Studio 2007.

Training en Workshops

De laatste tijd krijg ik veel aanvragen voor Delphi trainingen op locatie en workshops in mijn nieuwe cursusruimte in Helmond Brandevoort. Het betreft hier maatwerk trainingen, die niet alleen als voordeel hebben dat (alleen) de onderwerpen worden behandeld die gewenst zijn, maar ook dat er behalve de ontwikkelaars van het (ene) bedrijf, en geen andere deelnemers meegenieten, en het om die reden makkelijker is om bepaalde issues aan de orde te stellen en in alle rust te bespreken.

Wie interesse heeft in een dergelijke persoonlijke benadering, kan vrijblijvend contact opnemen om te kijken op welke manier ik ondersteuning kan leveren (als trainer, coach of consultant).

Ook Delphi support en ondersteuning per e-mail is mogelijk – zie de laatste pagina voor details.

Veel leesplezier met dit nummer van de Delphi OplossingsCourant, en schroom niet om op- of aanmerkingen te mailen of in mijn weblog achter te laten te <http://www.eBob42.com/weblog>.

Groetjes,

Bob Swart
Bob Swart Training & Consultancy

Het laatste Delphi Nieuws

Laten we beginnen met het laatste Delphi nieuws. En dan bedoel ik niet dat dit het einde is van Delphi, maar juist het begin van een nieuw hoofdstuk. Sinds gisteren is de .NET 2.0 versie van Delphi eindelijk aangekondigd, wat betekent dat het al over enkele dagen beschikbaar zal zijn. Er volgt dan meteen ook een Update #3 voor Delphi 2007 for Win32 en C++Builder 2007 om die op gelijke voet te brengen met de Delphi for Win32 en C++Builder die in de CodeGear RAD Studio 2007 zitten.

We hebben nu dus de keus uit de volgende verschillende versies:

☞ **CodeGear RAD Studio 2007**

- Delphi for Win32
- C++Builder
- Delphi for .NET 2.0

☞ **Delphi 2007 for Win32** (= hetzelfde als de Delphi for Win32 in de RAD Studio 2007)

☞ **C++Builder 2007** (= hetzelfde als de C++Builder in de RAD Studio 2007)

En voor .NET 1.1 compatibiliteit hoort ook de volgende nog in het lijstje:

☞ **Borland Developer Studio 2006**

- Delphi for Win32
- C++Builder
- Delphi for .NET 1.1
- C#Builder

Voor .NET 1.1 support zou je ook nog gebruik zou kunnen maken van **Turbo Delphi 2006 for .NET**.

De Studio edities komen in Professional, Enterprise of Architect editie, terwijl de "losse" Delphi 2007 for Win32 en C++Builder 2007 alleen in Professional en Enterprise edities beschikbaar zijn. Wie overigens al met Delphi 2007 for Win32 werkt, en geen behoefte heeft aan C++ of .NET, die heeft dus geen enkele reden om naar de CodeGear RAD Studio 2007 over te stappen – de Delphi for Win32 die daarin zit is dezelfde versie als Delphi 2007 for Win32 (na het uitvoeren van Update #3 dan).

Er zijn nog een heleboel ontwikkelaars die echter nog Delphi 7, 6, 5 of zelfs nog lager gebruiken. Hiervoor geldt dat de stap naar de laatste versie van Delphi for Win32 steeds groter wordt – niet alleen door de nieuwe IDE, maar ook door de third-party componenten die niet altijd meer (in dezelfde vorm) aanwezig zijn. Da's jammer, want er zijn in de loop der tijd aardig wat uitbreidingen in de Delphi taal en VCL componenten verzameling aangebracht. Ik krijg dan ook steeds vaker de vraag om te ondersteunen bij de migratie van een oudere versie van Delphi naar de laatste versie (en dan heb ik het nog niet eens over een mogelijke migratie naar .NET en VCL for .NET).

1.75 Miljoen Delphi Gebruikers

In dat kader was de recentelijke post van Michael Swindell (CodeGear VP of Products) erg interessant om te lezen. Zie ook <http://chrisbensen.blogspot.com/2007/08/delphi-user-statistics.html>. Letterlijk zei Michael het volgende:

Delphi has a developer population of approx 1.75 million users worldwide in 2006. Delphi has 15% IDE marketshare overall (7.7% ranked out of 100%) and is the #2 Windows IDE and the #4 IDE overall.

Verder gaf hij ook aan dat er over het algemeen meer exemplaren van C++Builder zijn verkocht, maar dat dat vaker Professional versies waren (tegen meer Enterprise of hogere versies bij Delphi).

CodeGear en “Subscriptions”

Toen Borland het .NET Framework ging ondersteunen, is dat gebeurd met een nieuwe IDE die van de grond af aan werd opgebouwd uit zowel Win32 als .NET onderdelen. Deze IDE kregen we als eerste te zien in het product C#Builder, en daarna in Delphi 8 for .NET. De derde incarnatie van de Borland Developer Studio ging als Delphi 2005 door het leven, en bevatte niet alleen de Delphi for .NET en C#, maar ook de Delphi for Win32 smaak. Delphi 2005 was dan ook voor Win32 ontwikkelaars de eerste zinvolle overstap na Delphi 7.

Bij de volgende versie (Borland Developer Studio 4.0) kwam er een vierde smaak bij in de vorm van C++Builder, waardoor dit de eerste nieuwe halte was voor C++Builder ontwikkelaars. BDS 4.0 werd onder de namen Delphi 2006, C++Builder 2006 en C#Builder 2006 verkocht, maar in feite kreeg men steeds hetzelfde product met vier "personalities" in handen. Behalve de complete studio, kun je ook een enkele smaak opstarten. Een nadeel van deze BDS oplossing die ik regelmatig hoorde, was dat het voor de zgn. "onderkant" van de markt (voor de thuisgebruikers en hobbyïsten) wel wat prijzig werd om bijvoorbeeld Delphi 2006 Professional aan te schaffen. Waarschijnlijk omdat er niet één maar vier talen in één omgeving werden aangeboden, was de prijs van de professional versie van Delphi ook gestegen. Gelukkig was ik niet de enige die dit hoorde, en er was een reactie van Borland in de vorm van de Turbo 2006 edities. Deze edities van de Borland Developer Studio bestaan maar uit één enkele smaak, kunnen niet uitgebreid worden (en ook niet op een machine neergezet worden die al een BDS 4.0 of andere Turbo versie hadden). Zowel in een betaalbare Professional editie als een gratis downloadbare Explorer editie bracht Borland ons Turbo Delphi 2006 for Win32, Turbo Delphi 2006 for .NET, Turbo C++ 2006 en Turbo C# 2006.

CodeGear

In de tussentijd was de Developer Tools Group binnen Borland (verantwoordelijk voor Delphi en C++Builder maar ook JBuilder en InterBase) bezig met een proces dat er uiteindelijk toe geleid heeft dat men onder de naam CodeGear verder gaat. CodeGear is nu een zelfstandig opererende organisatie, dochteronderneming van Borland, en daarmee zelf verantwoordelijk voor de uitgaven en inkomsten met betrekking tot de ontwikkeltools, wat tot vele positieve reacties heeft geleid zowel binnen als buiten CodeGear en de Delphi community. Als een van de nieuwe initiatieven heeft men bij CodeGear begin dit jaar een Delphi voor PHP uitgebracht (waarbij het de Delphi manier van werken is, en niet de Delphi taal voor PHP), en is men bezig met een ontwikkelomgeving (IDE) voor Ruby. Ook het opnieuw uitbrengen en updaten van de Delphi/C++Builder Roadmaps en de betrokkenheid bij de Delphi community worden als zeer positief ervaren.

CodeGear RAD Studio 2007

Met de ontwikkelingen rond CodeGear gaat de vijfde versie van de ontwikkelomgeving dan ook onder de naam CodeGear RAD Studio te boek. In het eerste kwartaal van dit jaar is Delphi 2007 for Win32 al uitgekomen als eerste onderdeel van deze CodeGear RAD Studio. In het tweede kwartaal kwam daar C++Builder 2007 bij. Ontwikkelaars kunnen een licentie van een van beide identiteiten aanschaffen en installeren. En ze zijn ook te integreren tot één omgeving, waardoor je zowel Delphi 2007 for Win32 als C++Builder binnen de CodeGear RAD Studio kunt krijgen. Hiervoor heb je dan twee licentie keys nodig.

Dat hoeft niet door zowel Delphi 2007 for Win32 als C++Builder 2007 aan te schaffen - het kan wel, maar is eigenlijk onnodig duur. Je kan nu namelijk ook de gehele CodeGear RAD Studio 2007 aanschaffen - en dan zit daar meteen ook Delphi for .NET 2.0 in.

Een alternatieve manier zou zijn geweest om bij de aanschaf van Delphi 2007 for Win32 aan Studio Level “Subscription” te doen, wat tevens een goed(kop)e manier is om je voor te bereiden om de dingen die nog komen gaan - een abonnement op je ontwikkelomgeving als het ware...

Subscription

Subscription kun je alleen afsluiten op hetzelfde moment dat je een licentie koopt. Wie dat niet doet, en een tijdje later alsnog een Subscription erbij wil kopen heeft helaas pech gehad, en zal tot de volgende aanschaf (nieuw of upgrade) moeten wachten om de Subscription erbij te nemen.

*Wanneer men een licentie koopt met Subscription, ontvangt men gedurende de looptijd van 12 maanden (optioneel verlengbaar met steeds weer 12 maanden) alle upgrades die vanaf het moment van aanschaf van de Subscription beschikbaar komen.
Daarnaast krijgt men ook nog 3 support incident calls per licentie.*

Let op: dat zijn dus upgrades (nieuwe versies). Updates (bug fixes) zijn altijd gratis voor geregistreerde gebruikers, en vallen dus ook niet onder de Subscription. Wees dus niet bang dat je geen bug fixes meer krijgt zonder Subscription.

Vlak voor het verstrijken van de Subscription periode (van 12 maanden) heb je steeds de mogelijkheid om de Subscription periode wederom met 12 maanden te verlengen (ik zal daar als CodeGear Reseller automatisch een herinnering voor naar de betreffende ontwikkelaar of onderneming sturen).

Als de Subscription onverhoopts toch verlopen is, of niet bij de initiële aankoop van de licentie is aangeschaft, is het helaas niet meer mogelijk om alsnog Subscription aan te schaffen. Als CodeGear Reseller zal ik dan dan ook altijd mijn uiterste best doen om de voordelen van Subscription te benadrukken, omdat je er niet achteraf nog aan kunt beginnen. Wie een nog actieve Subscription voor Delphi 2006 heeft, zal alle versies van Delphi 2007 en C++Builder 2007 inmiddels ook ontvangen hebben (de gehele CodeGear RAD Studio valt eronder). De prijs van Subscription is overigens gelijk aan 30% van de prijs van een nieuwe licentie voor de tool die je gekozen hebt.

Voor de "losse" Delphi 2007 for Win32 en C++Builder 2007 producten geldt dat een Subscription alleen geldig is voor de betreffende versie zelf (tenzij er destijds meteen een Studio Level Subscription is aangekocht – die is nu alleen nog bij de gehele CodeGear RAD Studio beschikbaar).

Dus wie alleen voor Win32 ontwikkelt, en alleen met Delphi (en niet met C++) die kan Delphi 2007 for Win32 aanschaffen en daarbij de (Identity Level) Subscription aanschaffen. Wie wel meerdere smaken wil gebruiken kan beter de hele CodeGear RAD Studio 2007 aanschaffen en daarbij de Studio Level Subscription nemen.

En als je slim bent verleng je iedere keer voor het aflopen van je Subscription deze voor weer een jaar (voor 30% van de nieuwprijs). Dat is goedkoper dan ieder jaar – of zelfs om het jaar – een upgrade te kopen, en alleen niet interessant als je minder vaak dan eens in de drie jaar een nieuwe versie koopt; dus als je bijvoorbeeld nog Delphi 7 of ouder gebruikt en geen interesse hebt in een nieuwe versie.

In alle andere gevallen biedt dit niet alleen een goedkope manier om een abonnement op de CodeGear RAD Studio te krijgen, het biedt ook een garantie wat betreft de prijs die ieder jaar verschuldigd is, en je kan er aan het eind van ieder jaar zonder probleem vanaf (verlengen is een optie – geen plicht).

Tot slot biedt het ook CodeGear een garantie betreffende hun inkomsten, die dan weer goed besteed kunnen worden aan de ontwikkeling van nieuwe versies en uitbreidingen van de CodeGear RAD Studio.

Kortom: ik raad iedereen aan om een Subscription te nemen!



Delphi 2007 voor Win32



Op 20 februari 2007 werd Delphi 2007 for Win32 aangekondigd, en op 19 maart – aan het eind van de eerste CodeRage virtuele conferentie – kwam Delphi 2007 for Win32 beschikbaar om aan te schaffen (en meteen via ESD te downloaden of op de DVD te wachten).

Delphi 2007 for Win32

Delphi 2007 for Win32 was het eerste onderdeel van de nieuwe CodeGear RAD Studio, en later dit jaar verscheen ook C++Builder tot tenslotte op 5 september 2007 de CodeGear RAD Studio 2007 compleet was met de Delphi for .NET 2.0 identiteit.

Delphi 2007 for Win32 kan los gebruikt worden, net als C++Builder 2007, maar deze twee kunnen ook geïntegreerd worden tot een CodeGear RAD Studio met twee persoonlijkheden. Wie de echte CodeGear RAD Studio koopt krijgt daar zelfs nog een derde persoonlijkheid bij in de vorm van Delphi for .NET 2.0 (compatible met .NET 3.0).

In dat opzicht is Delphi 2007 for Win32 dus niet te vergelijken met de Turbo Delphi edities. Turbo Delphi 2006 biedt slechts de mogelijkheid om één persoonlijkheid tegelijk op een machine te installeren, zonder die te kunnen uitbreiden. Dus wie Turbo Delphi for Win32 en Turbo Delphi for .NET eigenlijk tegelijk zou willen gebruiken, zou gewoon de CodeGear RAD Studio moeten hebben. Merk ook op dat Turbo Delphi pas later beschikbaar komt – nadat er al enkele updates voor de echte versie zijn uitgekomen. Turbo Delphi 2006 is nu te koop, terwijl Delphi 2007 for Win32 er ook al is. Toch verwacht ik Turbo Delphi 2007 for Win32 pas tegen het eind van het jaar – als het al komt – ruim nadat de andere onderdelen van de CodeGear RAD Studio ook al beschikbaar zijn gekomen.

CodeGear heeft de bedoeling om de RAD Studio te positioneren voor de professionele software ontwikkelaars, die vaak meer dan één variant nodig hebben (zoals Delphi voor zowel Win32 als .NET, of zowel Delphi and C++ voor Win32, of gewoon alles). De Turbo Delphi edities zijn meer voor de ontwikkelaar die aan één smaak voldoende heeft, en met de gratis Explorer versie kunnen daar ook de hobbyisten van profiteren.

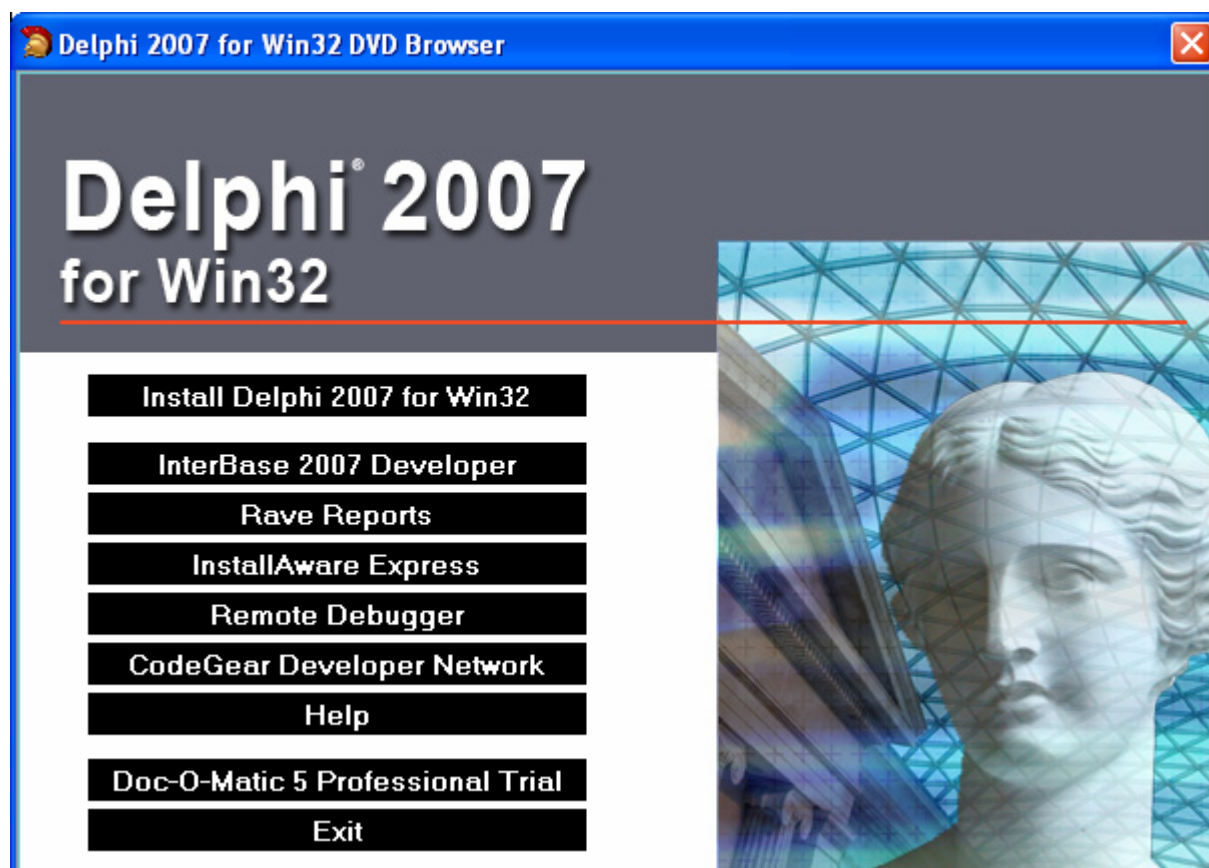
Delphi 2007 for Win32 is in twee smaken beschikbaar: Professional en Enterprise. Er is geen Architect smaak voor de Win32 versie, want Delphi Architect voegt met name de modelleertools toe aan Delphi Enterprise, en ECO is momenteel alleen beschikbaar voor .NET targets. Bij Delphi 7 Architect zat nog Bold van BoldSoft, maar dat bedrijf is inmiddels opgenomen in Borland (nu CodeGear), is gestopt met Bold en heeft de Enterprise Core Objects (ECO) functionaliteit toegevoegd. Grappig genoeg zijn de ontwikkelaars van ECO inmiddels ook weer uit CodeGear gestapt werken ze en onder de naam Capable Objects verder aan Enterprise Core Objects (dat nog steeds in de RAD Studio zit).

Installatie

Bij het installeren van Delphi 2007 for Win32 valt op dat versie 2.0 van het .NET Framework nodig is. Dat heeft al tot vele discussies geleid. Delphi 2007 for Win32 produceert geen .NET code. Maar de IDE zelf maakt wel gebruik van het .NET Framework versie 2.0. Behalve een ontwikkelplatform kun je het .NET Framework namelijk ook zien als een grote library waar al een heleboel zaken (op een goede en veilige manier) geregeld en beschikbaar zijn. En de ontwikkelaars bij CodeGear hebben daar dankbaar gebruik van gemaakt.

Tijdens de installatie – die overigens door de uitstekende InstallAware Wizard wordt verzorgd – wordt gecontroleerd of alle benodigdheden aanwezig zijn, en desgewenst gedownload of van de DVD gehaald. De DVD bevat bijna 4 GB aan informatie; alle smaken en talen van Delphi 2007 for Win32 staan erop, en op basis van je Product License Key wordt bepaald welke versie je op je systeem krijgt. Er is ook een kleinere installatie executable, en die zorgt er na het invoeren van de Product License Key voor dat alle benodigde onderdelen worden gedownload (daar heb je dus een goede internetverbinding voor nodig, anders duur het even).

Behalve Delphi 2007 for Win32, kun je vanaf de DVD ook InterBase 2007 Developer installeren, net als de nieuwe versie van Rave Reports, InstallAware Express, de Remote Debugger, een lokale kopie van de CodeGear Developer Network website, en een Doc-O-Matic 5 Professional Trial editie.



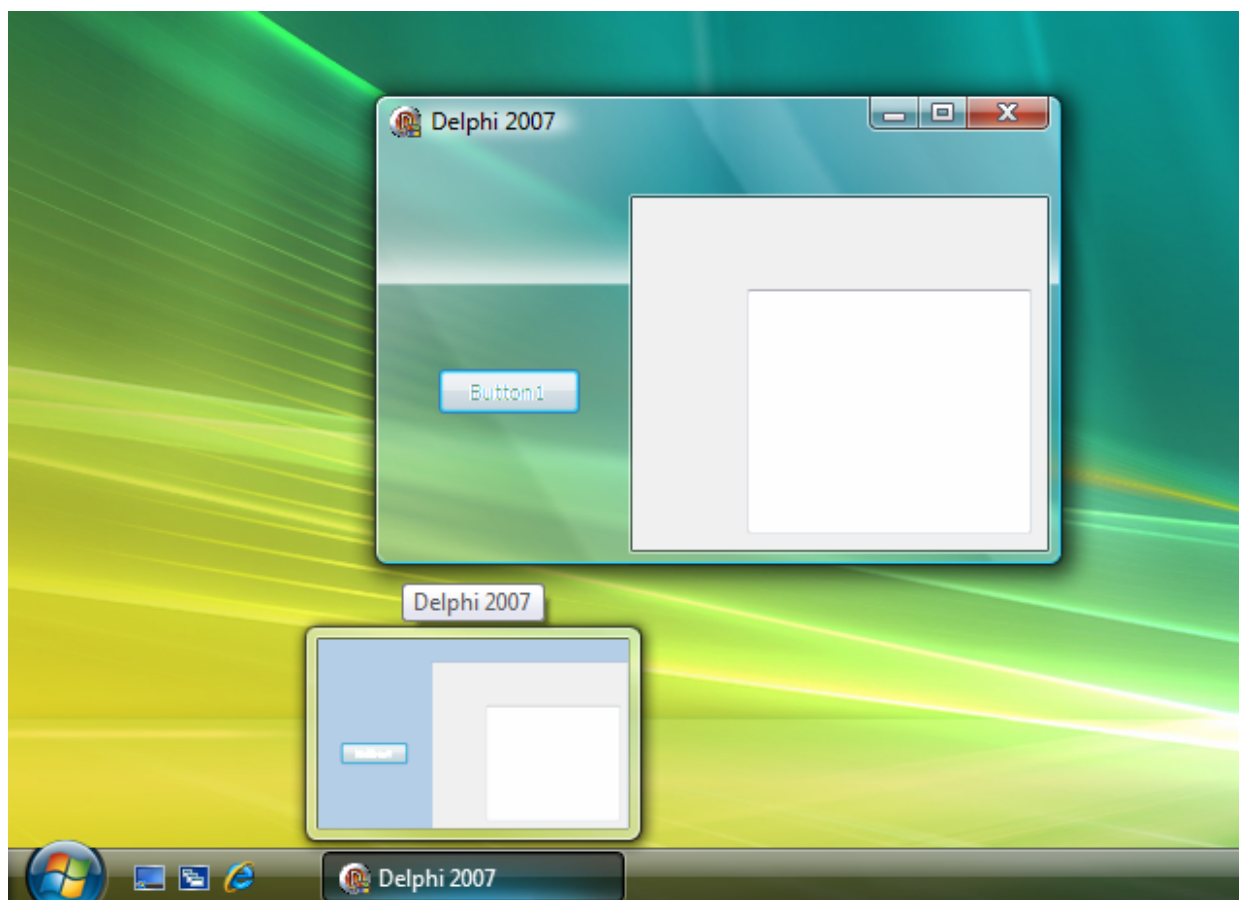
Wat is er nieuw?

In voorgaande versies van Delphi ging ik na installatie altijd op zoek naar het onderdeel in de helpfile met daarin de "What's New in Delphi X". De on-line help in de laatste drie versies van Delphi (Delphi 8 for .NET, Delphi 2005 en Delphi 2006) liet echter zelf link te wensen over. Met het uitkomen van Delphi 2007 for Win32 zouden we in ieder geval weer flink wat verbeteringen in de help moeten zien. De F1 toets werkt weer op (bijna) alle plaatsen, en ook de inhoud van de help is (iets) bruikbaar dan in de afgelopen jaren. Het is nog niet geweldig, maar er zit schot in.

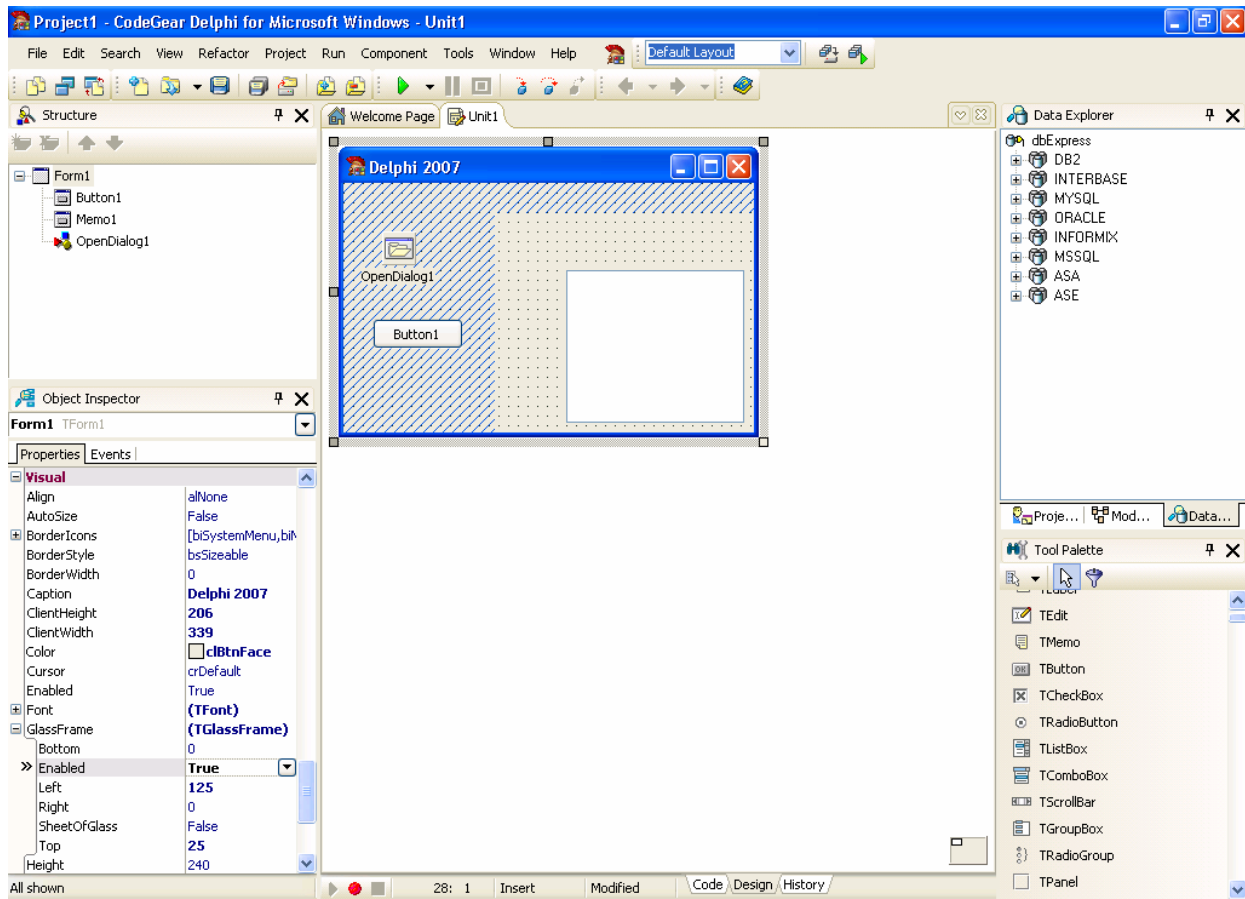
Windows Vista

Delphi 2007 for Win32 bevat native ondersteuning voor Windows Vista, en was daarmee de eerste ontwikkelomgeving met Vista ondersteuning. De Windows Vista ondersteuning bestaat o.a. uit een drietal Vista Dialog componenten, namelijk TFileOpenDialog, TFileSaveDialog en TTaskDialog. Deze drie componenten gebruiken de nieuwe dialogen die alleen onder Windows Vista beschikbaar zijn. Je kan ze ook onder andere versies van Windows tijdens design-time op je form zetten, maar kan ze alleen maar gebruiken tijdens run-time als je toepassing ook daadwerkelijk onder Windows Vista draait. Anders krijg je een EPlatformVersionException om je oren. Dat lijkt heel vervelend, en is het ook, behalve als je de readme.htm leest en ziet dat je de globale variabele UseLatestCommonDialogs moet gebruiken. Als je die op True zet, dan kun je gewoon de "normale" TOpenDialog en TSaveDialog blijven gebruiken, maar krijg je automatisch de nieuwe versies als je toepassing onder Windows Vista draait. Een beetje verwarrend is het wel, want er is in mijn ogen nauwelijks een reden om dus zelf ooit expliciet een TFileOpenDialog te gebruiken. De TOpenDialog is veel veiliger, en ziet er toch uit als een TFileOpenDialog als je onder Windows Vista draait.

CodeGear heeft dus niet de TFileOpenDialog nagebouwd zodat die ook onder Windows 2000 of XP er uitziet als onder Windows Vista - daar zou teveel tijd in gaan zitten. Er zou zeker teveel tijd gaan zitten in de simulatie van het "glas"-frame effect onder andere versies van Windows dan Windows Vista. Ieder TForm in Delphi 2007 heeft nu een GlassFrame property in de Visual category. Dit GlassFrame property bestuurt het gedrag van de titelbalk van je applicatie. Als je de Enabled subproperty op True zet, dan zal je titelbalk het glass effect vertonen (mits je onder Windows Vista draait, en mits Aero daar ook enabled is – anders zie je geen effect, maar ook geen foutmelding deze keer).



Behalve de Enabled subproperty, heeft de GlassFrame property ook een Bottom, Left, Right en Top property waarmee je de randen (van buiten naar het midden van je Form) van glas kunt laten maken. De subproperty SheetOfGlass maakt zelfs je hele Form als matglas. Dat is leuk voor even, maar natuurlijk niet echt werkbaar.



Alhoewel het GlassFrame dus alleen werkt onder Windows Vista met Aero enabled, zat er al een vergelijkbare feature in Delphi 7, namelijk de AlphaBlend Boolean en AlphaBlendValue properties. Door AlphaBlend op True te zetten kun je met de AlphaBlendValue de doorzichtigheid van je form aangeven op een schaal van 0 (niet meer zichtbaar) tot 255 (helemaal zichtbaar). Dat is vooral leuk als je in een for-loopje de waarde van 255 naar 0 laat aflopen en daarmee je scherm in het niets laat verdwijnen. Het is geen echt glas effect, maar komt toch aardig in de buurt. Een andere aardige uitbreiding is dat Delphi 2007 toepassing al van meet af aan theme support hebben (je hoeft niet langer zelf de XPManifest component op het form te zetten bijvoorbeeld).

AJAX

Wat meer relevant voor alle Windows gebruikers – maar dan met name voor web ontwikkelaars – is de AJAX ondersteuning die in Delphi 2007 zou zitten. Er wordt zelfs gesproken over “VCL for the Web”. Net als AJAX is dat echter een klein beetje “oude wijn in nieuwe zakken”, want “VCL for the Web” is in feite niets anders dan een nieuwe versie van IntraWeb met inderdaad ondersteuning voor AJAX technieken, maar niet iets geheel nieuws dat we nog nooit eerder gezien hadden. Delphi 7 gebruikers kregen al IntraWeb versie 5 meegeleverd, en met Delphi 2007 for Win32 zit IntraWeb versie 9.0 er bij. Het mooie is wel dat IntraWeb – of laten we het maar VCL for the Web – inderdaad AJAX mogelijkheden biedt zonder dat je als ontwikkelaar zelf alles moet afweten van HTML, JavaScript of stylesheets. Het mag wel, maar in principe zorgt VCL for the Web voor alle low-level details. In een toekomstig artikel zal ik hier uitgebreid op terugkomen.

MSBuild

Ook maakt Delphi 2007 nu gebruik van MSBuild, en dat heeft tot gevolg dat we pre-build en post-build commando's kunnen opgeven in de Project Options dialoog. Hiermee is het mogelijk om net als C++ compilers bepaalde zaken voor en na het build proces te regelen.

Delphi 2007 en DBX4

Een van de verbeteringen die we in Delphi 2007 for Win32 kunnen terugvinden, maar die niet meteen duidelijk zichtbaar zal zijn voor de meeste gebruikers, is DBX4. Versie 4 van de dbExpress data access architectuur. Het eerste wat ik mezelf afvroeg toen ik hoorde dat Delphi 2007 met DBX4 zou komen was: *“als Delphi 2007 met DBX4 komt, waar zijn dan de eerste drie versies van dbExpress gebleven?”*.

Gelukkig is het antwoord niet zo heel erg moeilijk te geven. De dbExpress data access techniek werd het eerst geïntroduceerd in Delphi 6, als cross-platform vervanger voor de BDE en SQL Links. In die tijd werkte Borland namelijk ook aan een versie van Delphi voor Linux, die later onder de naam Kylix zou verschijnen. De dbExpress componenten en database drivers zouden zowel voor Win32 als Linux beschikbaar komen, en die allereerste versie die bij Delphi zat is derhalve dbExpress versie 1.0. Van Kylix kwamen twee versies uit voor we aan Delphi 7 toekwamen. Kylix versie 1 kwam toch wel vrij laat (te laat?) op de markt, en was in de ogen van de meeste mensen nog niet helemaal af en veel te duur. Kylix versie 2 was al een stuk “af-er”, en werd ook een stuk goedkoper. De dbExpress bibliotheek bij Kylix had het versienummer 2 gekregen.

Er zijn mensen die het gevoel hebben dat Delphi 7 niet zoveel verschilt met Delphi 6, maar op een hoop punten heeft Borland de zaken toch beter geïmplementeerd. Vooral op het gebied van SOAP en Web Services (die zijn vrijwel volledig herschreven ten opzichte van wat er in de eerste release van Delphi 6 zat), maar ook dbExpress is met versienummer 2.5 in Delphi 7 weer van een nieuwe implementatie en nieuwe drivers voorzien. Dat versienummer vinden we vanaf Delphi 7 ook in de dbExpress DLLs en drivers zelf terug. Inclusief in de .NET versie van de dbExpress componenten die bij Delphi 8 for .NET zaten (de drivers zelf waren nog gewoon Win32 DLLs, maar de componenten waren VCL for .NET componenten). Ook Delphi 2005 maakt gebruik van dbExpress versie 2.5.

Delphi 2006 introduceerde dbExpress versie 3.0, en met de release van Delphi 2007 zitten we op dbExpress versie 4.0, ook wel DBX4 genaamd.

Waarom was deze opsomming belangrijk? Los van een historisch overzicht, is het ook relevant om te weten omdat dbExpress 4.0 – vanaf nu gewoon DBX4 genoemd – ook backwards compatible is met dbExpress 3.0 drivers. Maar niet met nog oudere drivers. Wie dus een third-party dbExpress driver heeft voor bijvoorbeeld FireBird of SQL Server (van bijvoorbeeld CoreLab – www.crlab.com), die doet er goed aan even te controleren welke versie van dbExpress deze driver is. Als dat een versie 3.0 driver is, dan kan hij nog gebruikt worden met Delphi 2007, anders (bij een lager versienummer) zul je op zoek moeten naar een nieuwe dbExpress driver om samen met Delphi 2007 te gebruiken. Inmiddels zijn die al beschikbaar, maar toen Delphi 2007 net uit was, was het toch even zoeken geblazen.

Delphi Source

Eerdere versies van dbExpress waren in een combinatie van C en C++ geschreven. DBX4 is echter van de grond af aan herschreven in Delphi. En compileer voor Win32 als .NET, zowel voor de componenten (in de dbExpress categorie) als de dbExpress drivers. Het voordeel is dat er nu per database maar één driver (source code project) onderhouden hoeft te worden, en niet twee (eentje voor Win32 en eentje voor .NET). Omdat onder .NET de Borland Data Provider (BDP) in de volgende versie van Delphi for .NET ook vervangen zal zijn door DBX4, betekent dat het verschil tussen vier drivers vroeger tegen één implementatie van de driver nu. En met in totaal zo’n acht verschillende databases die Delphi ondersteunt, kun je je voorstellen wat voor een verschil dat maakt voor het team. Niet alleen wat betreft onderhoud, maar ook het testen, certificeren en fixen van bugs. Wie de discussies in Quality Central een beetje gevolgd heeft m.b.t. de bugs in dbExpress en het fixen ervan, die kan zich wel een voorstelling maken van de voordelen die de single-source DBX4 vanaf nu met zich mee zal brengen.

Alles naar DBX4

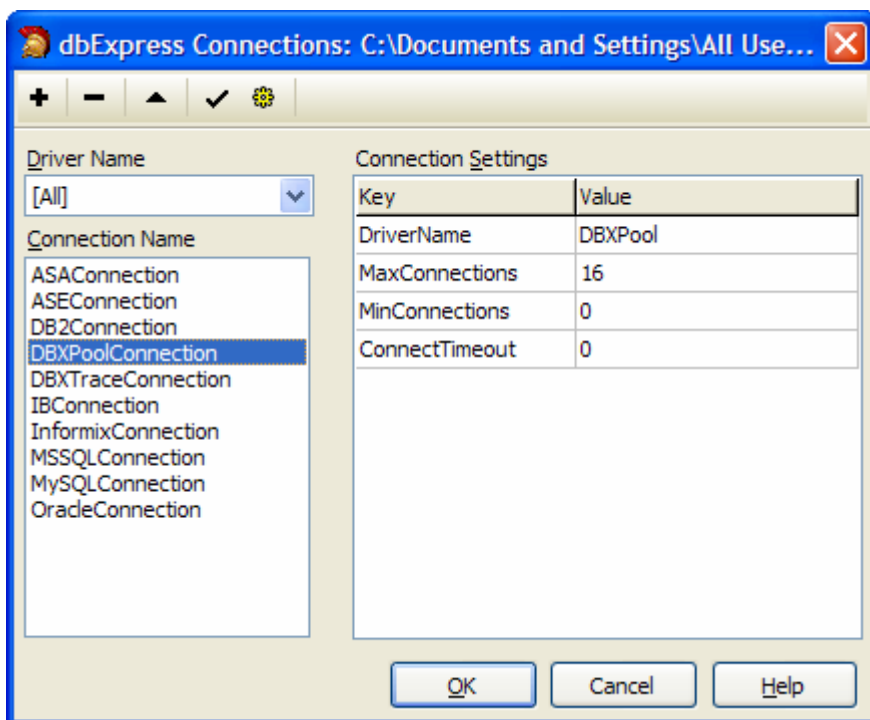
Het gevolg is wel dat CodeGear nu al zijn geld zet op DBX4. Vlak na het verschijnen van de eerste versie van dbExpress werd al gemeld dat de BDE bevroren was. De BDE zit echter nog steeds bij Delphi – ook bij Delphi 2007, en zal er niet uitgehaald worden, al was het maar om te voorkomen dat bestaande programma’s niet langer compileren. De BDE was voor gebruik van lokale tabellen; dBASE

en Paradox. Wie een echte DBMS wilde gebruiken, moest (in de tijd voor Delphi 6) gebruik maken van SQL Links, met daarbij de drivers voor de grote databases. Met het verschijnen van dbExpress is deze SQL Links “deprecated” gemaakt. En Delphi 7 is zelfs de laatste versie waarbij SQL Links wordt meegeleverd als onderdeel van de BDE. En inmiddels is het duidelijk dat er alleen nog database drivers voor DBX4 gemaakt zullen worden, dus wie een VCL toepassing maakt en met een recente database wil werken, zal of met DBX4 aan de slag moeten, of op zoek naar een third-party data access driver voor deze database.

Voor nieuwe projecten is het niet moeilijk om DBX4 te gebruiken; de dbExpress componenten die bij Delphi 6 zaten werken nog vrijwel allemaal ongewijzigd in de nieuwe DBX4 versie (er zijn enkele verschillen; daar kom ik zo nog op terug). Lastiger is het als we een bestaande BDE / SQL Links toepassing hebben, bijvoorbeeld geschreven in Delphi 5 of Delphi 3 nog (ze komen nog voor!), en we deze willen omzetten naar Delphi 2007. Geen overstap van Win32 naar .NET, maar puur een overstap van een hele oude versie van de Delphi IDE naar een modernere versie, plus de overstap van BDE / SQL Links naar dbExpress (DBX4) en een nieuwe driver voor de betreffende database. Deze migratie is niet altijd even eenvoudig, en daar zal ik de volgende keer meer over schrijven. Deze keer wil ik het echter met name over DBX4 zelf hebben, en de nieuwe mogelijkheden ten opzichte van de dbExpress die we kennen sinds Delphi 6.

Drivers

Om te zien welke drivers er in DBX4 zitten, hoef je alleen maar een TSQLConnection component op een Form of Data Module te zetten, en de Connections Editor te starten. Behalve de drivers voor Adaptive Sybase SQL Anywhere 9 (ASA), Sybase 12.5 (ASE), IBM DB2 UDB 8.x (DB2), InterBase 2007 en 7.5 (IB), Informix 9.x, MS SQL Server 2005 en 2000, MySQL 5.0.27 en 4.0.24, en Oracle 10g, zijn er nog twee nieuwe drivers te zien die we nog niet eerder gezien hadden: de DBXPoolConnection en DBXTraceConnection.



De DBXPoolConnction kan gebruikt worden voor connection pooling, en de DBXTraceConnction voor het traceren. Geen van beide maken echter zelf gebruik van een database – daar hebben ze een “echte” connection voor nodig. Wat dat betreft zijn zowel de DBXPoolConnection en de DBXTraceConnection een soort “semi” connecties, die ook wel onder de naam delegate connections door het leven gaan. Ze kunnen gebruikt worden om een “echte” connection van extra mogelijkheden te voorzien.

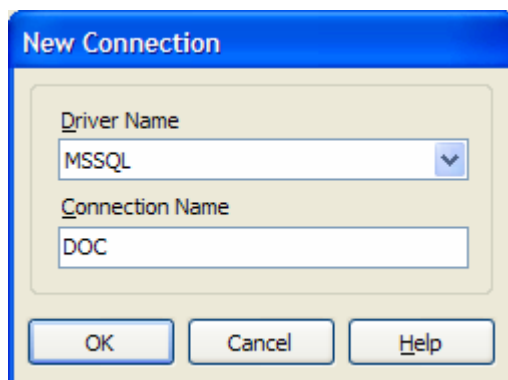
Connection Pooling

Laten we allereerst eens de DBXPoolConnection onderzoeken. Deze kan gebruikt worden om connection pooling te realiseren. Hierbij wordt de connection in een DBX4 pool bijgehouden – deze mogelijkheid is dus onafhankelijk van het feit of de database zelf ook connection pooling zou ondersteuning of niet.

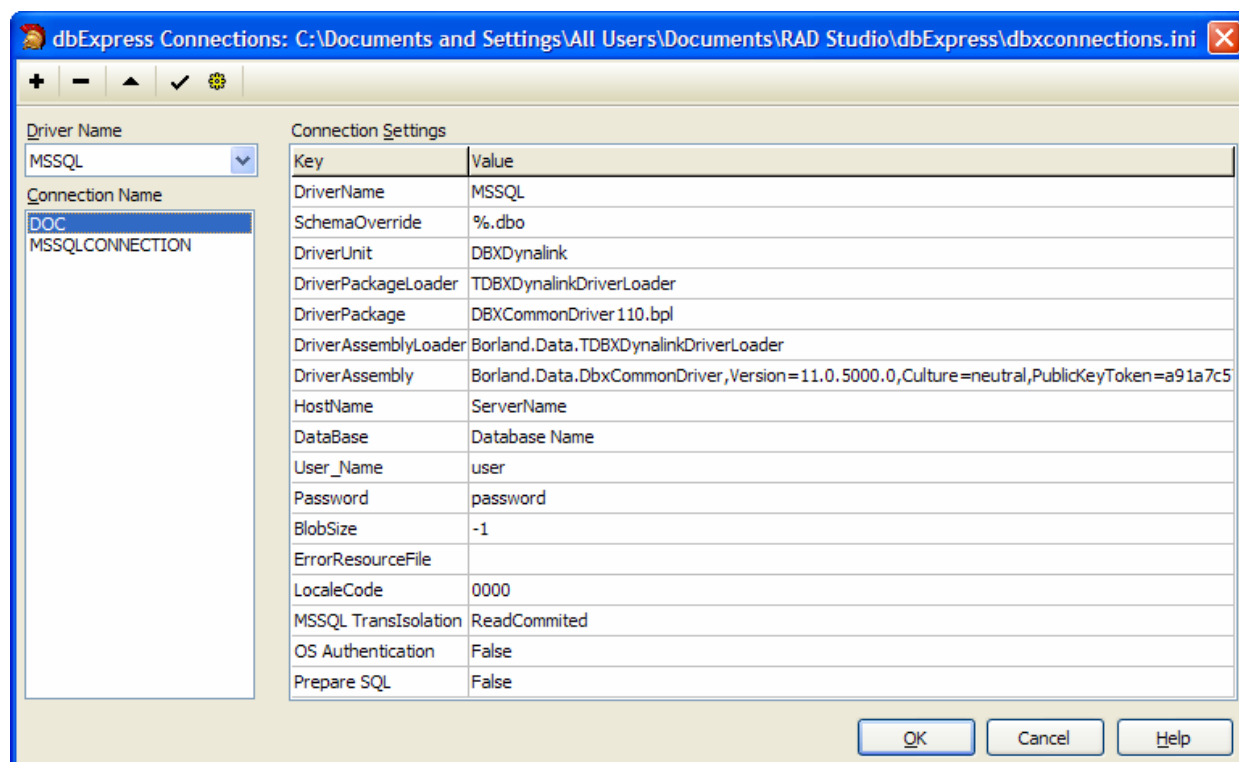
```
[DBXPoolConnection]
DriverName=DBXPool
MaxConnections=16
MinConnections=0
ConnectTimeout=0030
```

Nieuwe Connectie

Ik zal nu een voorbeeld geven van de opbouw van een delegate connectie. Allereerst gebruik ik de Connections Editor (zie het eerste screenshot) om een nieuwe connectie te maken. In de dialoog die volgt, kies ik voor het type MSSQL (omdat ik met een SQL Server database wil werken), en als naam van de connectie geef ik "DOC" op.



De Connections Editor laat dan in de lijst met connections naast de default MSSQLConnection ook de nieuwe DOC connection zien.



Vergeleken met de default MSSQLConnection valt het wel op dat de DOC connection een aantal extra attributen laat zien, waarvan het nut niet meteen duidelijk is. Met name de DriverUnit, DriverPackageLoader, DriverPackage, DriverAssemblyLoader en DriverAssembly attributen zijn niet zichtbaar in de MSSQLConnection, en worden eigenlijk al op het niveau van de MSSQL driver bepaald. Dus wat mij betreft kunnen ze eigenlijk wel weg (en je moet ze in ieder geval niet aanpassen zonder dat je weet wat je doet).

De DriverPackage is hiervan de enige die relevant is voor Win32 toepassingen, en laat zien dat we de DBXCommonDriver110.bpl package moeten deployen met onze DBX4 toepassingen. Het gekke is echter dat ik een package met de naam DBXCommonDriver110.bpl helemaal niet kan vinden op mijn machine. In plaats daarvan moet ik de DBXCommonDriver100.bpl gebruiken.

Delegate Drivers

Alhoewel de Connections Editor wel enkele attributen laat zien waar we eigenlijk niks mee kunnen doen, valt het me op dat er helaas (nog) geen manier is om aan te geven wat we als delegate connection willen gebruiken. Hiervoor moeten we namelijk een attribuut met de naam "DelegateConnection" hebben om een waarde te kunnen geven. Deze key kunnen we niet toevoegen in de Connections Editor zelf. In plaats daarvan moeten we de Params property van de TSQLConnection component gebruiken om de lijst met parameters te editen, en hier kunnen we dan een DelegateConnection parameter toevoegen met dan als waarde DBXPoolConnection of DBXTraceConnection. Dit wijzigt overigens alleen de waarde van de DOC Connection die gebruikt wordt door dit betreffende TSQLConnection component.

Als we een meer blijvende wijziging willen aanbrengen in de DOC Connection, zullen we de file dbxconnections.ini in C:\Documents and Settings\All Users\Documents\RAD Studio\dbExpress directory moeten bewerken (deze filenaam is terug te vinden in de caption van de dbExpress Connections Editor).

De DOC Connection connection ziet er in de dbxconnections.ini als volgt uit:

```
[DOC]
DriverName=MSSQL
SchemaOverride=% .dbo
DriverUnit=DBXDynalink
DriverPackageLoader=TDBXDynalinkDriverLoader
DriverPackage=DBXCommonDriver110.bpl
DriverAssemblyLoader=Borland.Data.TDBXDynalinkDriverLoader
DriverAssembly=Borland.Data.DbxCommonDriver,Version=11.0.5000.0,...
HostName=ServerName
DataBase=Database Name
User_Name=user
Password=password
BlobSize=-1
ErrorResourceFile=
LocaleCode=0000
MSSQL TransIsolation=ReadCommitted
OS Authentication=False
Prepare SQL=False
```

Merk op dat de lijst ook nog de DriverUnit, DriverPackageLoader, DriverPackage, DriverAssemblyLoader en DriverAssembly attributes omvat die we eerder in het screenshot zagen. Aangezien deze vijf attributen ook gevonden kunnen worden in de [MSSQL] entry in de dbxdrivers.ini file, kunnen ze verwijderd worden uit de [DOC] sectie!

En als we toch brzig zijn in de dbxconnections.ini kunnen we meteen de DelegateConnection key toevoegen met DBXPoolConnection of DBXTraceConnection als waarde:

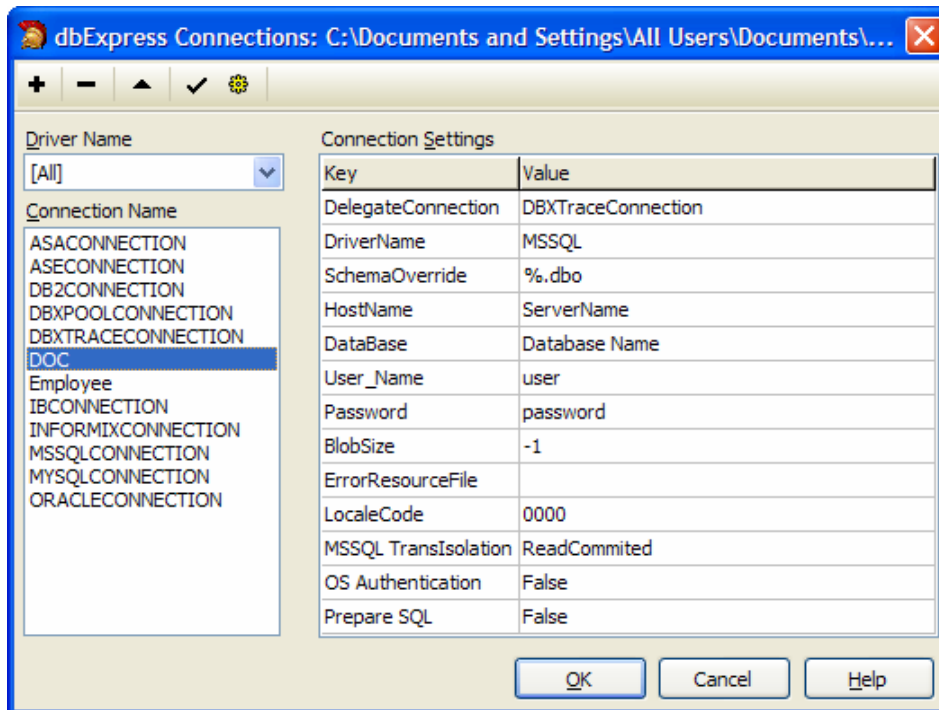
```
[DOC]
DelegateConnection=DBXTraceConnection
DriverName=MSSQL
SchemaOverride=% .dbo
HostName=ServerName
DataBase=Database Name
User_Name=user
Password=password
```

```

BlobSize=-1
ErrorResourceFile=
LocaleCode=0000
MSSQL TransIsolation=ReadCommitted
OS Authentication=False
Prepare SQL=False

```

Binnen Delphi moet ik de Connections Editor nog een keer openen om de parameters van de TSQLConnection component te “verversen”. Ik kan dan meteen de daadwerkelijke server, database, user en password informatie invullen.



De DOC Connection werkt dus met de SQL Server Northwind database, maar maakt als delegate connectie ook gebruik van de DBXTraceConnection (eentje die al bij Delphi 2007 zit), en hiermee kunnen we op eenvoudige wijze de verbinding met de Northwind database traceren (indien nodig natuurlijk – dit is niet iets wat je altijd “aan” wilt hebben staan).

Tracing

De DBXTraceConnection zelf heeft ook een aantal attributen, waarvan er enkele in commentaar zijn geplaatst in de dbxconnections.ini, om aan te geven wat de mogelijkheden een beetje zijn. Ik heb zowel de TraceFile als de TraceFlags uit het commentaar gehaald, met het volgende resultaat:

```

[DBXTraceConnection]
DriverName=DBXTrace
TraceFile=c:\temp\bcctrace.txt
TraceFlags=PARAMETER;ERROR;EXECUTE;COMMAND;CONNECT;TRANSACTION;BLOB;MISC;VENDOR;READER;DRIVER;
OAD;METADATA
TraceDriver=false

```

Als je geen filename opgeeft, dan is de trace output te zien in een console window. Dat werkt overigens alleen als je je Delphi toepassing met {\$APPTYPE CONSOLE} hebt aangemerkt (of de Console optie in je Project Options hebt aangevinkt). Anders krijg je een I/O Error 105 error. Dat gebeurt overigens ook als je geen TraceFile hebt opgegeven en tijdens design-time een verbinding probeert te openen, want Delphi zelf is natuurlijk geen console toepassing.

De output van de DBXTraceConnection delegate driver bestaat niet uit al-dan-niet-leesbare log statements, maar bestaat uit een overzicht van de Delphi statements die op het betreffende moment daadwerkelijk worden uitgevoerd.

Dat is een stuk handiger en je ziet – als Delphi ontwikkelaar – dan ook meteen wat er gebeurt en waar het eventueel misgaat.

```
Log Opened =====
{CONNECT      } ConnectionC1.Open;
{COMMAND      } CommandC1_1 := ConnectionC1.CreateCommand;
{COMMAND      } CommandC1_1.CommandType := 'Dbx.SQL';
{COMMAND      } CommandC1_1.RowsetSize := 20;
{COMMAND      } CommandC1_1.CommandType := 'Dbx.SQL';
{COMMAND      } CommandC1_1.Text := 'select * from Employees';
{PREPARE      } CommandC1_1.Prepare;
{COMMAND      } ReaderC1_1_1 := CommandC1_1.ExecuteQuery;
{COMMAND      } CommandC1_2 := ConnectionC1.CreateCommand;
{COMMAND      } CommandC1_2.CommandType := 'Dbx.MetaData';
{COMMAND      } CommandC1_2.Text := 'GetIndexes "Northwind"."dbo"."Employees" ';
{COMMAND      } ReaderC1_2_1 := CommandC1_2.ExecuteQuery;
{READER       } { ReaderC1_2_1 closed. 2 row(s) read }
{READER       } FreeAndNil(ReaderC1_2_1);
{COMMAND      } FreeAndNil(CommandC
```

Als je daarnaast ook de log statements op driver niveau wilt zien, dan moet je de TraceDriver attribute op True zetten. Dit zijn niet-Delphi achtige log statements, maar die kunnen ook nuttig zijn.

Transactions

Als je bestaande toepassingen die gebruik maken van dbExpress om wilt zetten naar Delphi 2007 en DBX4, dan kan het zijn dat je soms een melding krijgt dat methods niet meer bestaan. Met name op het gebied van transacties krijg je die melding voor de StartTransaction, Commit en Rollback methodes. Die zijn deprecated, aldus de melding. Helaas wordt er niet bij verteld wat je dan wel moet gebruiken, maar na enig zoekwerk kan ik dat hier vertellen.

We moeten nu de nieuwe methodes BeginTransaction, CommitFreeAndNil en RollBackFreeAndNil van de TSQLConnection component gebruiken, bijvoorbeeld als volgt:

```
DBXTransaction := SQLConnection1.BeginTransaction(TDBXIsolations.ReadCommitted);
try
  // do your work...
  SQLConnection1.CommitFreeAndNil(DBXTransaction);
except
  SQLConnection1.RollBackFreeAndNil(DBXTransaction);
  raise
end;
```

Daarnaast was de unit dbExpress vroeger gevuld met de core van dbExpress, maar deze unit is nu ook deprecated en in plaats daarvan moeten we de units DBXCommon en/of SqlExpr gebruiken in de uses clause.

Deployment

Een nadeel van DBX4 in de eerste versie van Delphi 2007 for Win32 is het feit dat we niet langer dbExpress drivers kunnen meelinken met onze Delphi VCL toepassingen. De MidasLib unit is er nog wel, maar de dbExpress unit niet meer, dus moeten de drivers zelf gedeployed worden. Dit zou na Update #3 opgelost kunnen zijn, maar daarover later meer.

Behalve de dbExpress driver, moeten we soms ook de dbxdrivers.ini en dbxconnections.ini file deployen met onze toepassing, met name als we de DBXTraceConnection of DBXPoolConnection gebruiken. Geen van beide delegate drivers zijn namelijk verbonden aan een TSQLConnection component gevuld met parameters, dus zal het DBX4 framework de informatie uit de .ini files moeten halen. Voordeel is wel dat je die .ini files bijna leeg kunt laten, op de [DBXTrace] en [DBXTraceConnection] secties na dan.

Tot slot moeten we de DBXCommonDriver100.bpl package deployen (als we met packages werken), en dan ook tenminste de rtl100.bpl en dbrtl100.bpl, want die twee worden gebruikt door DBXCommonDriver100.bpl (en de Delphi 2007 versies van deze packages zijn verschillend van de Delphi 2006 versies – tot zover het “non-breaking” van de Delphi 2007 release dus).

Conclusie

Delphi 2007 heeft in DBX4 weer een nieuwe benadering voor databases, volledig herschreven in Delphi, maar op component niveau vrijwel ongewijzigd (op transactie support na). Er zijn genoeg voordelen en nieuwe features, alleen op deployment gebied mag het nog wat beter worden, zeker aangezien de drivers zelf in Delphi zijn geschreven. De ontwikkelingen zijn nog niet afgelopen, en met de volgende release (Delphi for .NET 2.0), zal ook DBX4 in een Win32 en .NET smaak terug te vinden zijn, met nog wat extra mogelijkheden- zie ook verderop in deze DOC voor vast wat details.

De volgende keer zal ik een aantal voorbeelden geven om van de BDE over te stappen op DBX4. Een noodzakelijk maar helaas niet geheel triviale aangelegenheid. Tot dan!

Updates

Er zijn inmiddels al twee updates verschenen van Delphi 2007 for Win32. De eerste kwam vrij snel, en de tweede kwam gelijk met de release van C++Builder 2007. De derde update staat er nu ook aan te komen, en bevat alle fixes en nieuwe Win32 features die ook in de CodeGear RAD Studio 2007 zitten. Dus wie niet overstapt naar de CodeGear RAD Studio 2007, zal toch kunnen profiteren van alle fixes en updates.

Prijzen

Delphi 2007 for Win32 is beschikbaar in een Enterprise en Professional editie. Onderstaande prijzen zijn exclusief BTW (nog steeds 19%) en gelden voor de Electronic Software Delivery (ESD) edities, die je kunt downloaden en kunt installeren met een bijbehorend serienummer dat je ontvangt.

Delphi 2007 for Win32 Enterprise

| | |
|-------------------------------|------------------|
| New User | € 1.724,= |
| Upgrade | € 1.124,= |
| Subscription Delphi for Win32 | € 525,= per jaar |

Delphi 2007 for Win32 Professional

| | |
|-------------------------------|------------------|
| New User | € 774 |
| Upgrade | € 324 |
| Subscription Delphi for Win32 | € 235,= per jaar |

Voor C++Builder 2007 gelden dezelfde prijzen. Maar je krijgt dus maar één product voor deze prijs. Wie ze allebei wil hebben – of allemaal, inclusief de Delphi for .NET 2.0 identiteit – is goedkoper uit om de hele CodeGear RAD Studio aan te schaffen. Ter vergelijking volgen hier de prijzen voor Enterprise en Professional:

CodeGear RAD Studio 2007 Enterprise

| | |
|----------------------------------|------------------|
| New User | € 1.974,= |
| Upgrade | € 1.334,= |
| Subscription CodeGear RAD Studio | € 650,= per jaar |

CodeGear RAD Studio 2007 Professional

| | |
|----------------------------------|------------------|
| New User | € 934 |
| Upgrade | € 414 |
| Subscription CodeGear RAD Studio | € 315,= per jaar |

Special Aanbieding: Wie een CodeGear ontwikkelomgeving bij mij aanschaf in combinatie met de Subscription, krijgt bij mij (gedurende de looptijd van de Subscription) 50% korting op de aanschaf van Delphi cursusboeken in PDF formaat die ik aanbied van <http://www.eBob42.com/courseware> (nieuwe Delphi 2007 boeken komen later dit jaar, en zijn dan ook met korting beschikbaar).

Stuur e-mail naar Bob@eBob42.com voor meer informatie of om een bestelling te plaatsen.

Windows Vista en ASP.NET 1.1

Windows Vista is het nieuwste besturingssysteem van Microsoft, en wordt tegenwoordig al op veel nieuwe machines standaard meegeleverd. Inclusief nieuwe laptops. Voor het installeren van Delphi 7 of 2006 onder Windows Vista heb ik eind vorig jaar een artikel op m'n website gezet te <http://www.drBob42.com/examines/examin84.htm>. Wie vervolgens met Delphi aan de slag wil, loopt helaas al snel tegen een aantal kleine problemen aan. De meeste hiervan worden veroorzaakt door de nieuwe strenge regels onder Windows Vista, en dan met name de User Account Control.

Er is echter een probleem dat niet zo makkelijk op te lossen is, en dat betreft het debuggen van ASP.NET 1.1 toepassingen met Delphi (versie 8 for .NET, 2005 of 2006) onder Windows Vista en IIS 7. Onder Windows Vista mag je eigenlijk niet als Administrator werken, maar moet je onder een "gewone" user account aan de slag. Het probleem is dan dat de C:\Inetpub\wwwroot directory niet aangepast kan worden – dat mag je alleen als Administrator.

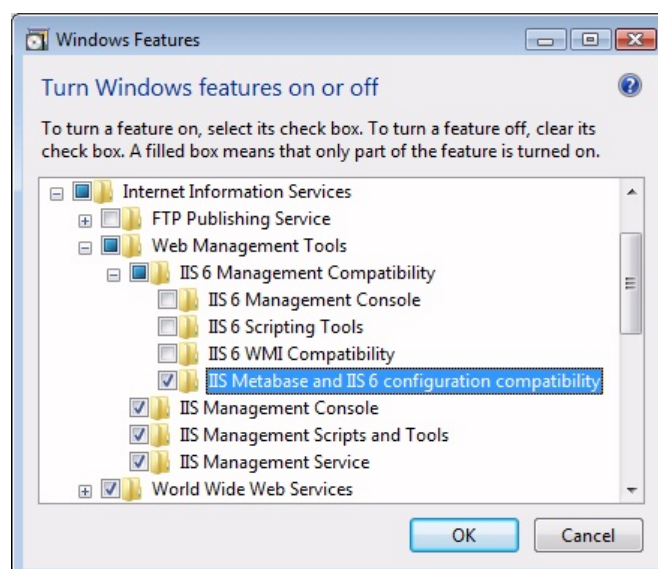
Er zijn verschillende oplossingen voor dit probleem. Het meest eenvoudig is om je ASP.NET projecten gewoon ergens anders neer te zetten op de schijf (waar je wel mag schrijven), of om aan de directory C:\Inetpub\wwwroot de permissies toe te voegen om een gewone user erin te laten schrijven. Dat laatste is natuurlijk niet zo veilig. Gelukkig is er nog een derde alternatief, maar die vertel ik straks pas.

Internet Information Services (IIS) 7

Eerst is het tijd om even stil te staan bij de versie van Internet Information Server (IIS) die we onder Windows Vista vinden. Dat is namelijk een nieuwe versie: IIS 7. Windows NT kwam destijds met IIS 4, Windows 2000 had IIS 5, en Windows XP zit op IIS 5.1. Windows Server 2003 introduceerde IIS 6, en is daarmee op dit moment de meest gebruikte versie van IIS op Windows-based web servers. En nu hebben we IIS 7 dat met Windows Vista komt, terwijl Windows Vista alles behalve een web server deployment platform is. Da's dus lastig, als je tegen IIS 7 moet ontwikkelen, maar onder IIS 6 gaat deployen (het zal nog tot Windows Server 2008 duren voordat IIS 7.1 er voor een Windows (web) server versie is, die bruikbaar is als web server deployment omgeving).

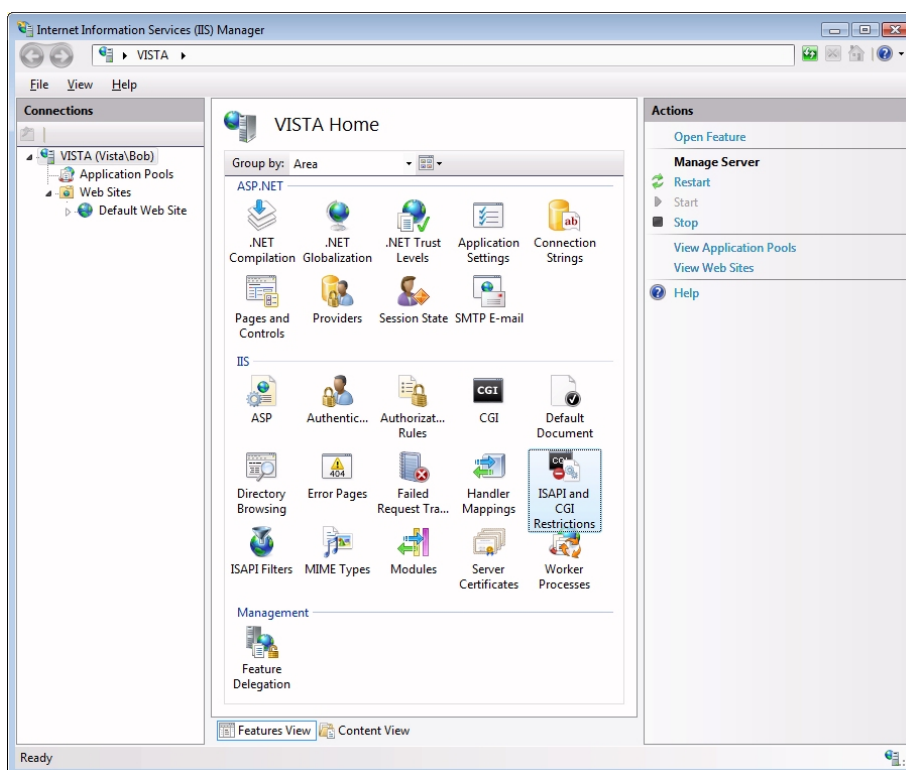
Om onder Windows Vista toch nog met ASP.NET 1.1 onder IIS 6 compatibility te kunnen werken moeten we via het Control Panel en de Programs groep naar de "Turn Windows features on or off" feature gaan (we krijgen dan de User Account Control dialog te zien overigens).

In de Windows Features dialoog moeten we dan bij de Internet Information Services, Web Management Tools, IIS 6 Management Compatibility boom de optie "IIS Metabase and IIS 6 configuration compatibility" aanzetten:

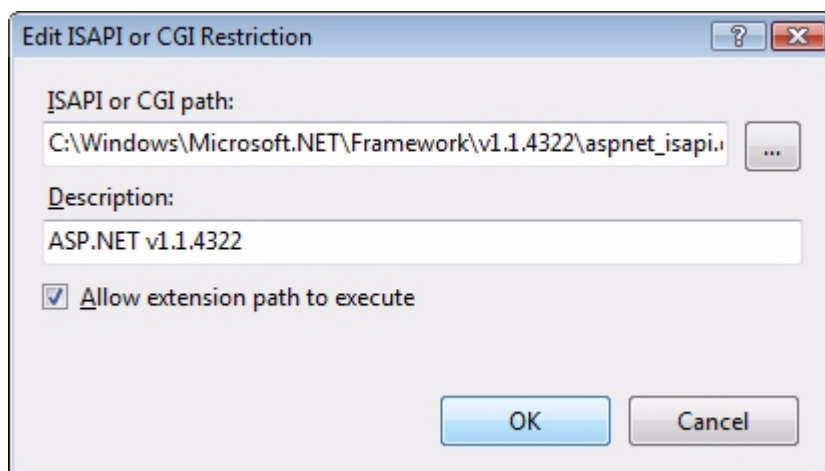


Dit zorgt ervoor dat de IIS metabase en compatibility laag beschikbaar komt om metabase calls te laten omgaan met de nieuwe IIS 7.0 configuration store.

Maar dan zijn we er nog niet. Vervolgens moeten we ASP.NET 1.1 ook nog aanzetten (let op: ik ga ervanuit dat Delphi en ASP.NET 1.1 zelf al geïnstalleerd zijn). Default staat ASP.NET 1.1 namelijk uit onder Windows Vista. Om dit te verhelpen moeten we naar de "Internet Information Services (IIS) Manager" gaan, die te vinden is in de Administrative Tools (of in de Computer Management Console). Hier moeten we de machine zelf selecteren - VISTA (Vista\Bob) in mijn geval – en dan in de Features View dubbelklikken op de "ISAPI and CGI Restrictions" (zoals misschien weinig mensen weten, wordt ASP.NET in feite mede gerealiseerd door ISAPI filters)

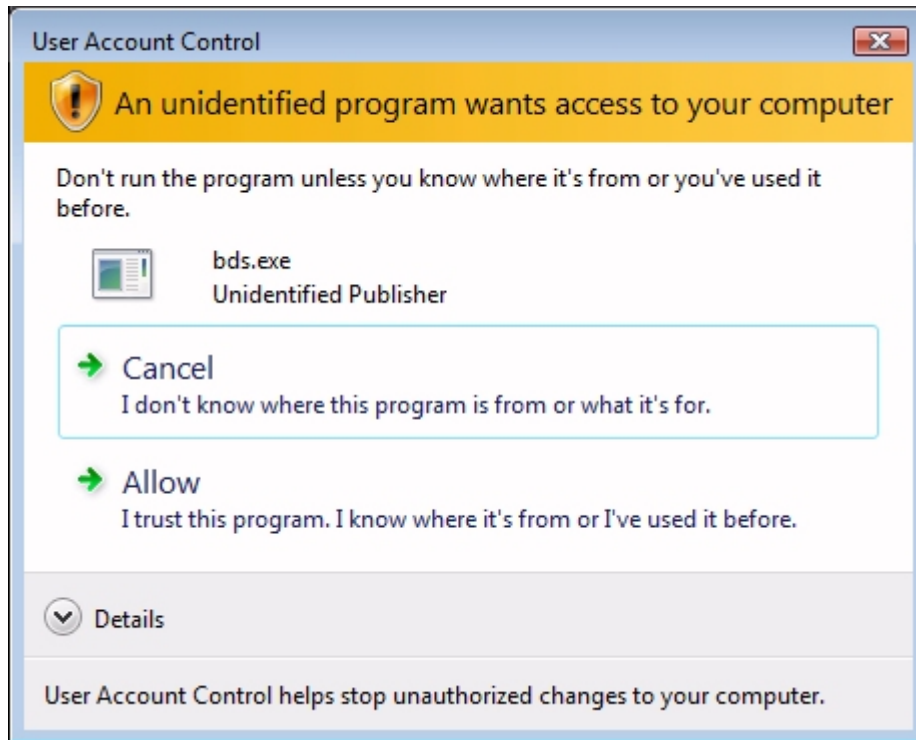


Het scherm dat je dan krijgt laat zien dat ASP.NET v1.1.4322 "Not Allowed" is onder Windows Vista (zelfs als het goed geïnstalleerd is), terwijl ASP.NET v2.0.50727 wel toegestaan is. De oplossing is dan eenvoudig: we moeten ASP.NET v1.1.4322 ook toestaan hier:



Vervolgens kunnen we Delphi starten – voor ASP.NET 1.1 is dat Delphi 8 for .NET, Delphi 2005 of Delphi 2006 (met de CodeGear RAD Studio 2007 maak je ASP.NET 2.0 toepassingen, en daarbij kan CodeGear RAD Studio 2007 wel goed met IIS 7 omgaan).

Belangrijk is dat we Delphi starten met de "Run as administrator" optie, waarna we wederom een User Account Control dialog te zien krijgen:

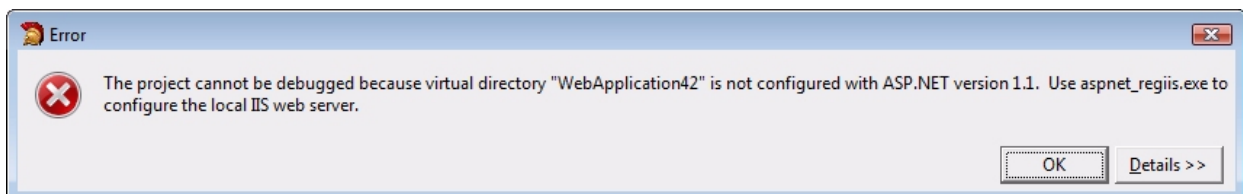


Maar zonder Delphi als Administrator te draaien, zijn we niet in staat om nieuwe ASP.NET 1.1 virtual directories te maken in de C:\Inetpub\wwwroot directory (tenzij je al een van de eerdere "oplossingen" had toegepast).

Nu kunnen we een nieuwe ASP.NET toepassing maken, om te kijken of we deze kunnen draaien en debuggen. Doe dus *File | New - ASP.NET Web Application - Delphi for .NET*, en geef de naam in van je ASP.NET toepassing. Zet een Button, ListBox en TextBox op de web form, en schrijf één regel code in de OnClick van de Button, als volgt:

```
procedure TWebForm1.Button1_Click(sender: System.Object; e: System.EventArgs);  
begin  
    ListBox1.Items.Add(TextBox1.Text);  
end;
```

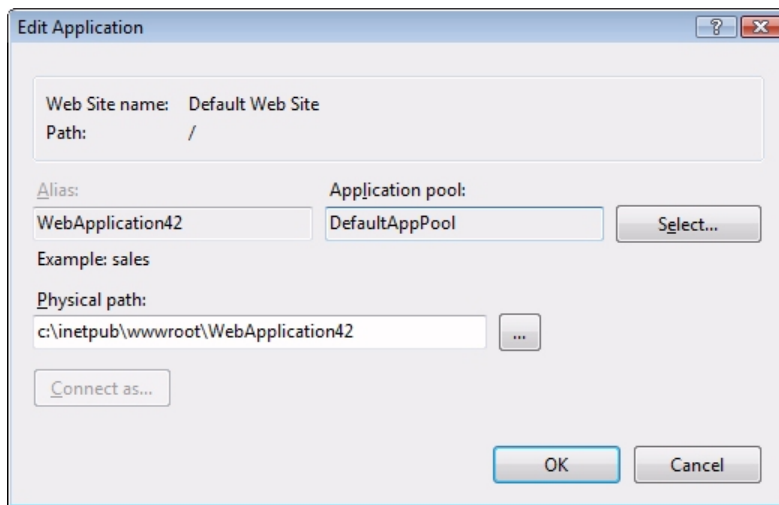
Zet nu een breakpoint op deze regel code, en doe *Run | Run* om het project te debuggen. Helaas levert dit een foutmelding op die beweert dat de virtual directory niet als ASP.NET versie 1.1 geconfigureerd is.



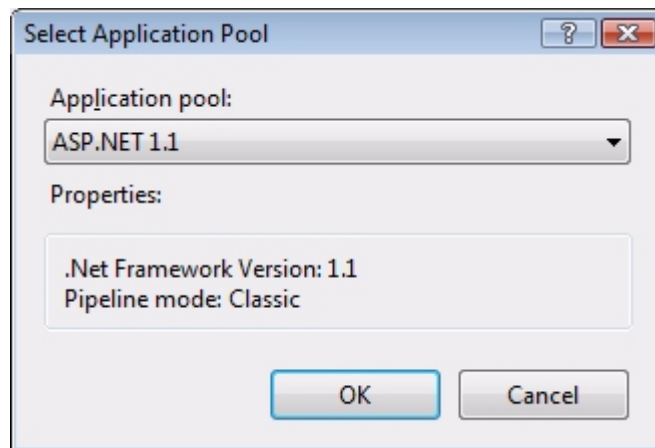
Dit is inderdaad het geval: default worden alle virtual directories als ASP.NET 2.0 aangemerkt, en zullen we ze handmatig moeten toevoegen aan de ASP.NET 1.1 application pool.

Hiervoor hebben we wederom de Internet Information Services Manager nodig. Deze keer moeten we naar de Server, Web Sites, Default Web Site, en daarna onze virtual directory, en daar op de Basic Settings klikken (aan de rechterkant van het window).

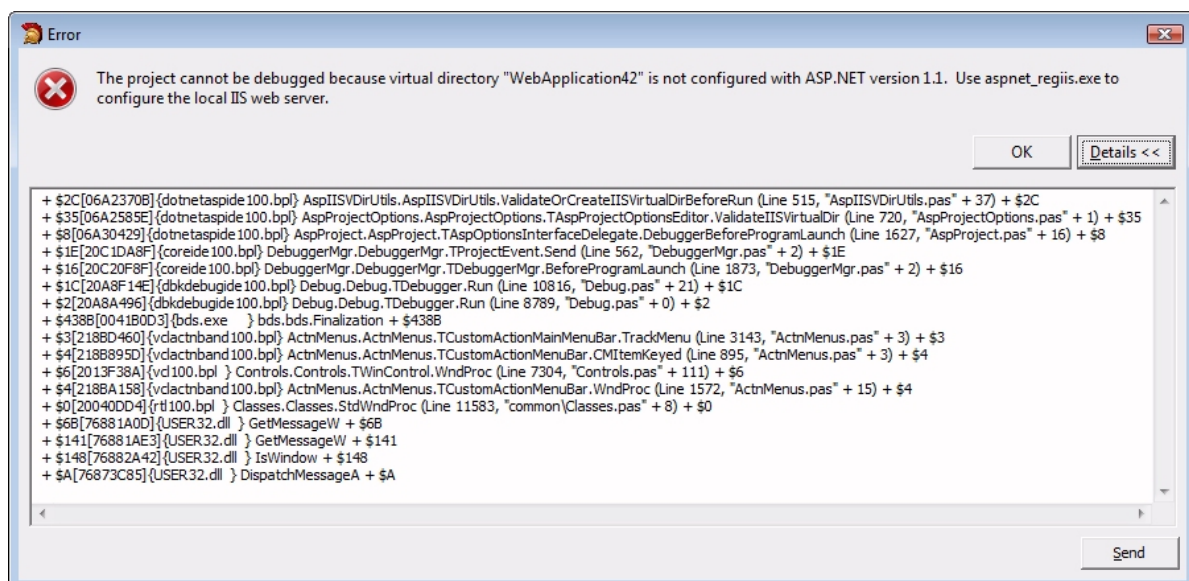
Nu krijgen we een dialoog die laat zien dat de nieuwe ASP.NET toepassing in de DefaultAppPool zit, en die is voor ASP.NET 2.0.



Via de Select knop kunnen we dit veranderen in de ASP.NET 1.1 Application Pool.

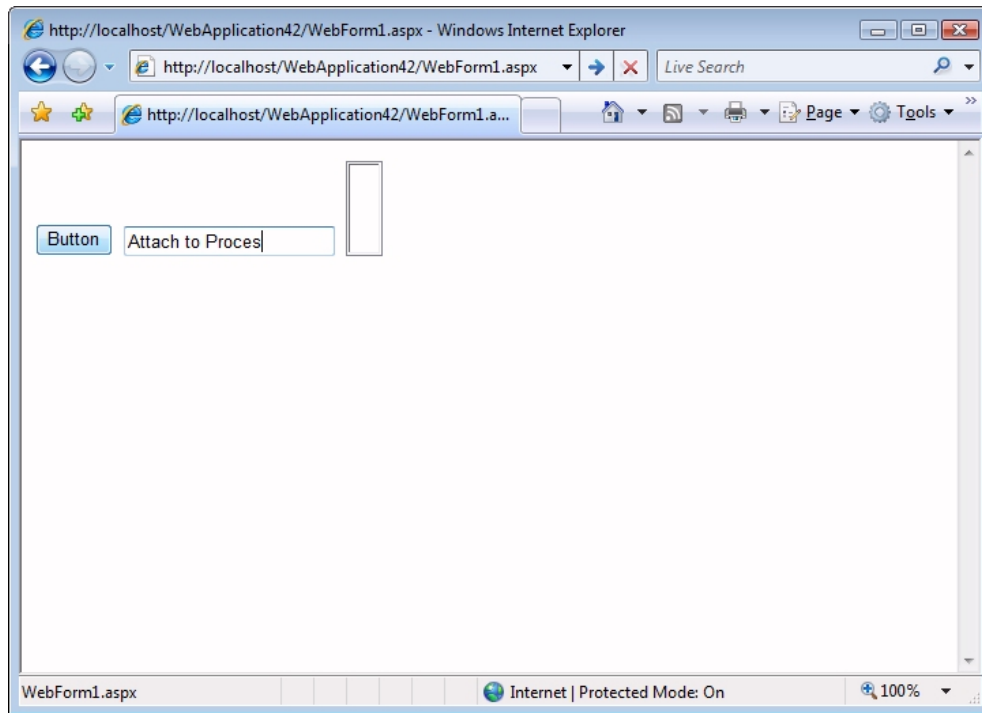


Nu weten we zeker dat de toepassing in de virtual directory als ASP.NET 1.1 geconfigureerd is. Helaas is Delphi daar nog niet van overtuigd, want we krijgen nog steeds een foutmelding bij het runnen, zelfs na een *Run | Run without Debugging*.

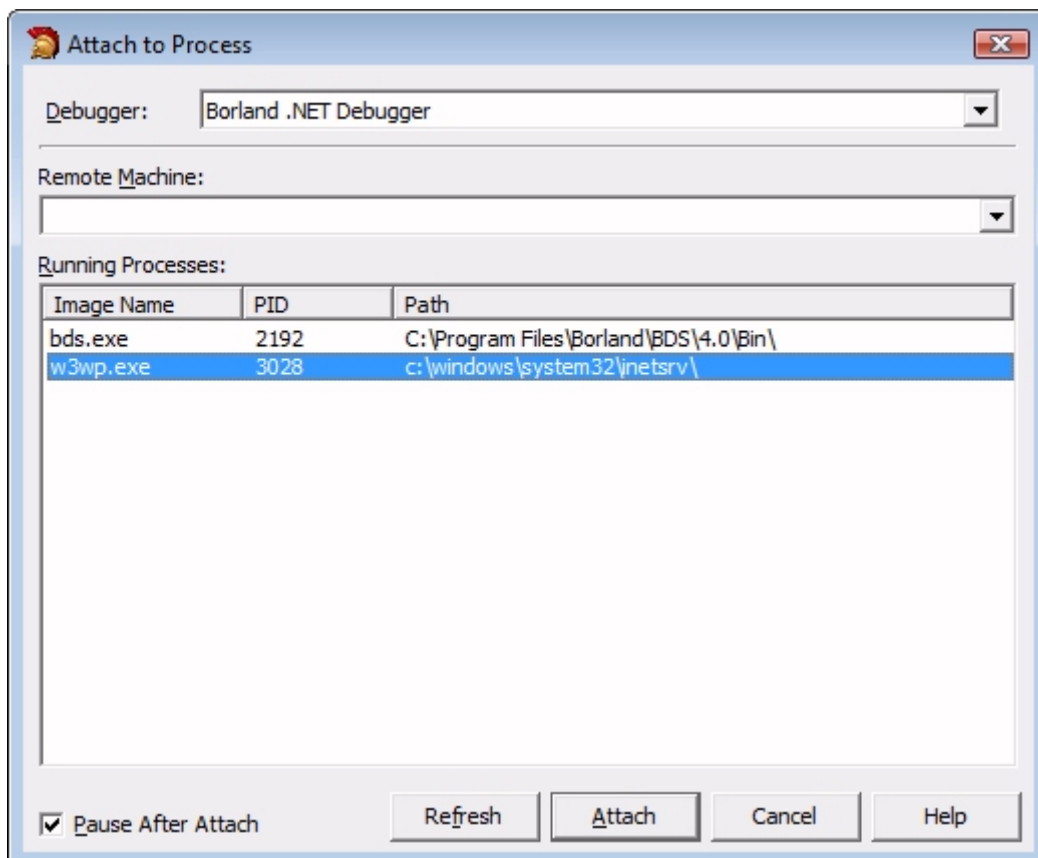


Delphi blijft van mening dat de virtual directory niet goed is. En vanuit de IDE kunnen we het project dus niet draaien.

De enige oplossing die overblijft (totdat CodeGear dit probleem heeft opgelost in Delphi 2006), is om de URL van de ASP.NET toepassing zelf in een browser in te tikken:

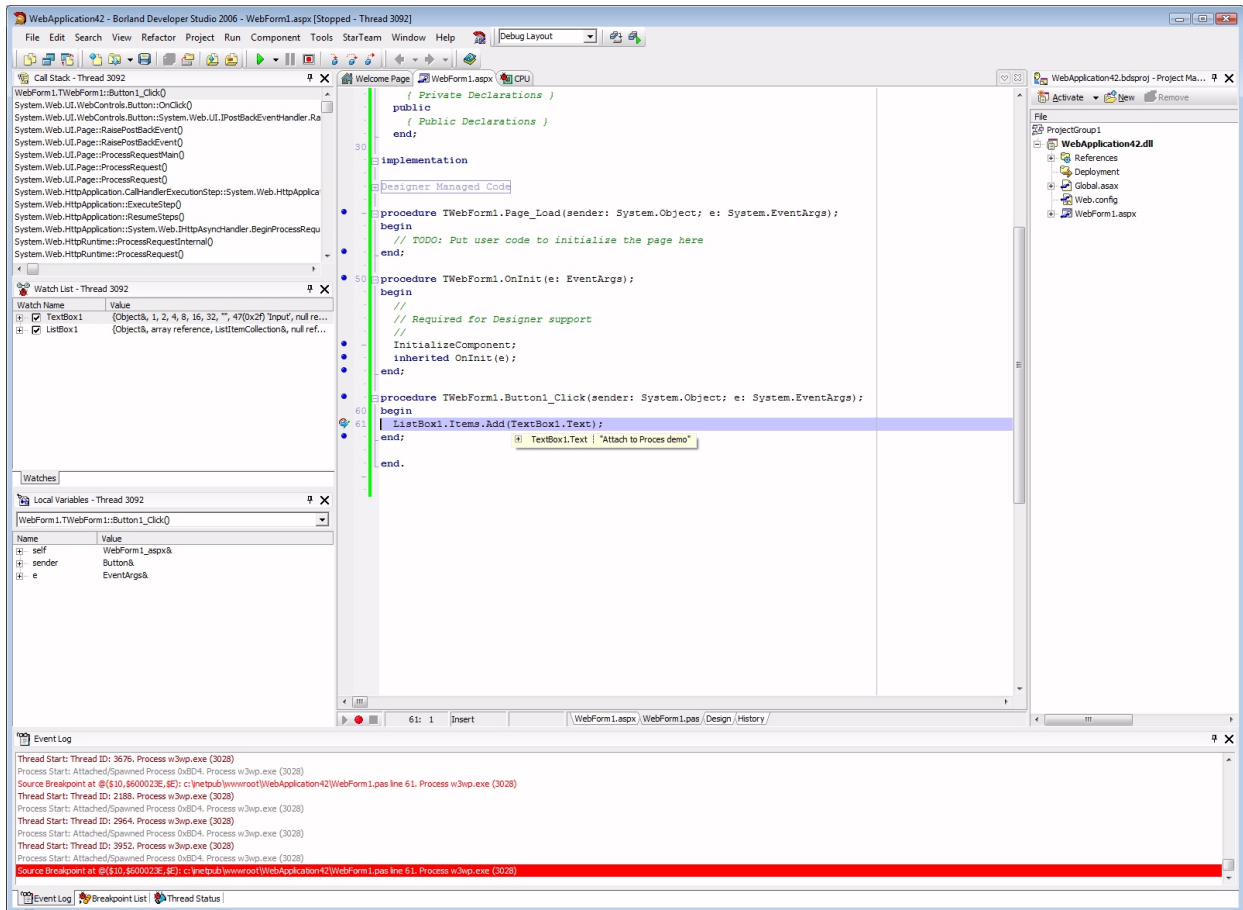


Vervolgens kunnen we in Delphi via de *Run | Attach to Process* op zoek naar de `w3wp.exe` en daar een Attach mee doen.



Let op: dit kan alleen voor het ASP.NET 1.1 Worker Process. Als je de virtual directory niet in de ASP.NET 1.1 Application Pool hebt gezet, zal deze nog als ASP.NET 2.0 draaien, en dan krijg je een foutmelding ("Could not create process") als je vanuit Delphi 2006 daar een Attach naar probeert te doen.

Als de Attach gelukt is, krijg je binnen Delphi de Debug Desktop te zien, en kun je je breakpoints zetten. Onderstaande screenshot laat het moment zien waarop het breakpoint getriggerd is, en we de tekst "Attach to Process demo" (die ik in de browser had ingetikt) te zien krijgen in de tooltip evaluatie van de TextBox1.Text.



Het is niet geweldig, maar het werkt. Vergeet niet om na afloop van de debug sessie weer *Run | Detach from Program* te doen.

Het probleem met het debuggen van Delphi 2006 ASP.NET 1.1 toepassingen onder Windows Vista en IIS 7 is inmiddels aangemeld bij CodeGear in hun Quality Central omgeving, als een rapport met nummer 50800, zie <http://qc.codegear.com/wc/qcmain.aspx?d=50800> voor details.

De workaround die voorlopig wordt voorgesteld is om met Cassini te werken, maar dat is niet echt een optie in mijn ogen, omdat je dan helemaal niet aan het ontwikkelen en testen bent met een server die lijkt op waar je uiteindelijk op gaat deployen.

Kortom: het laatste woord is hier nog niet over gezegd of geschreven. Hou mijn weblog in de gaten voor verdere ontwikkelingen en eventuele oplossingen. Of overweeg gewoon om Windows Server 2003 en IIS 6 op je ontwikkelmachine te zetten in afwachting van Windows Server 2008 en IIS 7.1, maar laat Windows Vista links liggen (in ieder geval voor ASP.NET toepassingen).

CodeGear RAD Studio 2007



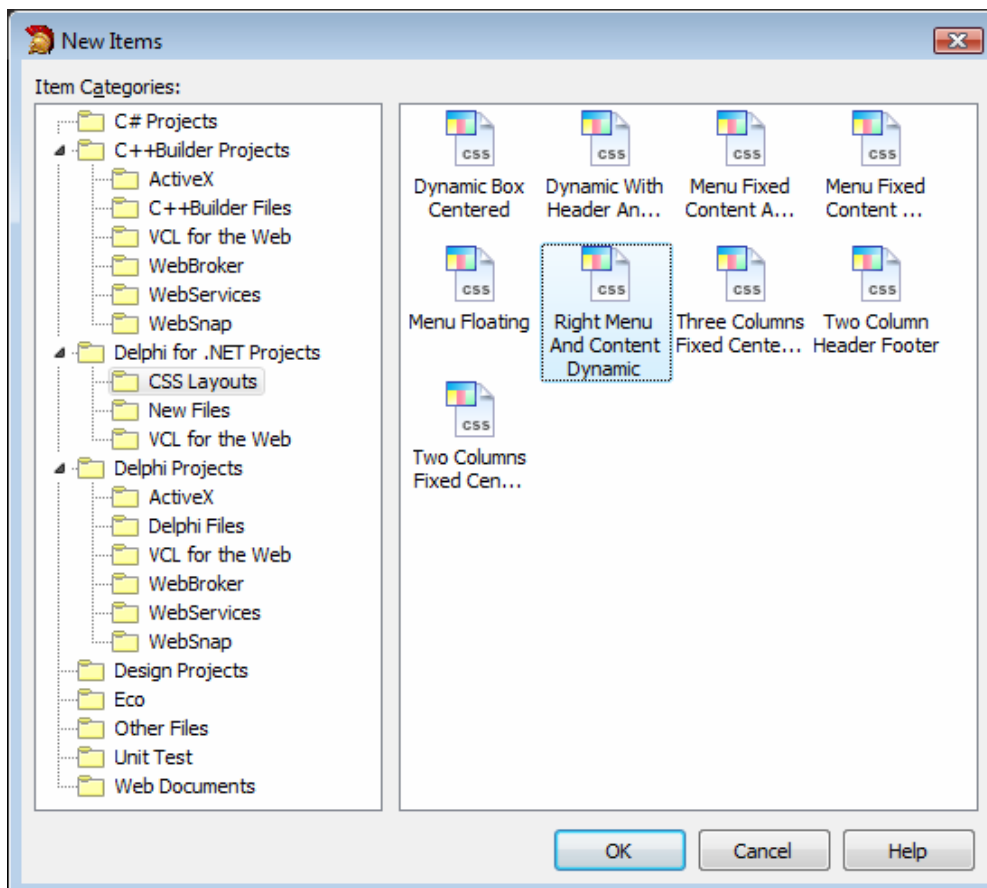
Woensdag 5 september was het zover: de CodeGear RAD Studio 2007 werd officieel aangekondigd, nadat er een lange tijd onder de codenaam Delphi Highlander aan gewerkt is. Over enkele dagen zal de CodeGear RAD Studio 2007 daadwerkelijk beschikbaar zijn, alsmede de Update #3 voor Delphi 2007 for Win32 en C++Builder. In de tussentijd volgt hier vast een korte impressie van enkele van de nieuwe features.

Delphi for .NET 2.0

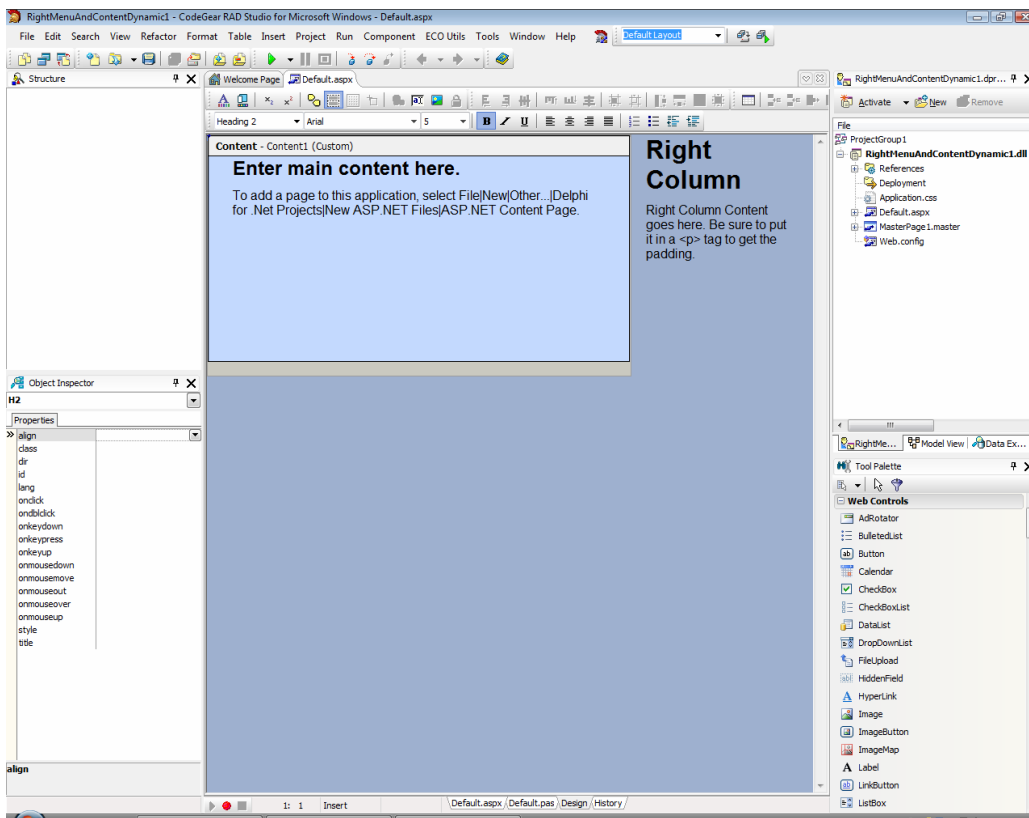
Een van de zaken die nieuw is in de CodeGear RAD Studio is natuurlijk de Delphi for .NET 2.0 identiteit. Deze is niet los te krijgen als "Delphi 2007 for .NET 2.0", maar uitsluitend als onderdeel van de hele RAD Studio aan te schaffen.

ASP.NET 2.0

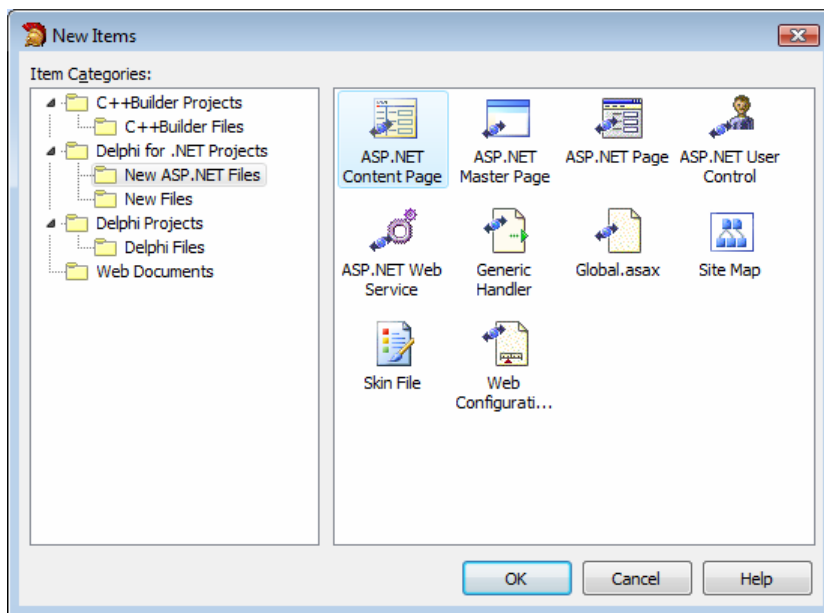
Met Delphi for .NET 2.0 verdwijnt de ondersteuning voor WinForms, maar krijgen we eindelijk ondersteuning voor ASP.NET 2.0 en de mogelijkheden die daarbij worden aangeboden. Master pages, Membership, Roles en verschillende Providers, en nog veel meer. Een van de handige nieuwe features is terug te vinden in de CSS Layouts sectie van de Delphi for .NET Projects in de Object Repository:



Het kiezen van een van deze CSS Layout items resulteert in een nieuw ASP.NET 2.0 Web Forms applicatie, waarbij er meteen een MasterPage is toegevoegd met een layout zoals die grofweg in de omschrijving van het item is aangegeven. Dus voor de "Right Menu And Content Dynamic" icon krijgen we het volgende resultaat:



Je kunt nu aan de MasterPage aanpassingen maken, en vervolgens een of meerdere Content Pages toevoegen die afgeleid zijn van de Master Page. Een beetje als Form Inheritance in de VCL.



Let ook op de andere New ASP.NET Files items in de Object Repository, zoals de Site Map en Skin File, die ook nieuw zijn in CodeGear RAD Studio 2007. Meer wil ik op dit moment niet over ASP.NET 2.0 schrijven, maar hou m'n website in de gaten voor nieuwe artikelen (of kom langs voor een workshop).

Generics

Een andere feature van .NET 2.0 die nu in Delphi for .NET zit staat bekend onder de naam Generics. Onder Generics worden in feite geparametriseerde types bedoeld; een type met een parameter, waardoor het bijna type-loos is. Voor de syntax hierbij maakt Delphi for .NET gebruik van de <T> syntax, net als C++ templates dus als het ware.

Een voorbeeld van een eenvoudig generic Delphi array is als volgt:

```
type
  Some<T> = Array of T;
```

Hierbij geeft de T aan over wat voor soort array we het eigenlijk willen hebben. We kunnen bijvoorbeeld een Integer, String maar ook een DataSet array maken op deze manier.

Dit kan als volgt:

```
type
  IntegerArray = Some<Integer>;
  StringArray = Some<String>;
  DataSetArray = Some<DataSet>;

var
  X: SomeIntegers;
```

Nadeel van dit eenvoudige voorbeeld is dat het maar een array is – wel data, maar verder geen (extra) functionaliteit, dus eigenlijk weinig toegevoegde waarde aan het gebruik van een Generic type hier.

Een leuker voorbeeld is de Queue, waarbij we wel functionaliteit toevoegen, en niet van te voren hoeven op te geven wat voor type elementen we in de Queue stoppen.

De definitie van een Generic TQueue type kan als volgt zijn:

```
type
  TQueue<AnyType> = class
  protected
    FList: System.Collections.Generic.List<AnyType>;
  public
    constructor Create;
    destructor Destroy; override;
    function Count: Integer;
    function AtLeast(count: Integer): Boolean;
    function Push(const itemT: AnyType): AnyType;
    function Peek: AnyType;
    function Pop: AnyType;
  end;
```

De implementatie van de TQueue methods zoals de Constructor en Destructor maken ook gebruik van de nieuwe <T> syntax, als volgt:

```
constructor TQueue<AnyType>.Create;
begin
  inherited Create;
  FList := System.Collections.Generic.List<AnyType>.Create
end;

destructor TQueue<AnyType>.Destroy;
begin
  FList.Free;
  inherited Destroy
end;
```

Voor de implementatie van de onderliggende structuur om de generic types in op te bergen, kunnen we gebruik maken van een reeds bestaand Generic type uit het .NET 2.0 Rramework (waarmee ik laat zien dat we niet alleen zelf nieuwe Generic types kunnen maken, maar ook bestaande Generic types zoals de System.Collections.Generic.List kunnen importeren en gebruiken).


```

function TQueue<AnyType>.Count: Integer;
begin
    Result := FList.Count
end;

function TQueue<AnyType>.AtLeast(count: integer): Boolean;
begin
    Result := FList.Count >= count
end;

function TQueue<AnyType>.Push(const itemT: AnyType): AnyType;
begin
    FList.Insert(0, itemT);
    Result := itemT
end;

function TQueue<AnyType>.Peek: AnyType;
begin
    Result := FList[FList.Count-1]
end;

function TQueue<AnyType>.Pop: AnyType;
begin
    Result := Peek;
    FList.RemoveAt(FList.Count-1)
end;

```

Het declareren en gebruiken van deze TQueue kan vervolgens in Delphi for .NET 2.0 op de manier en met de syntax die ik al eerder liet zien (zie volgende pagina):

```

type
    TStringQueue = TQueue<string>;

var
    S: TQueue<string>; // or TStringQueue;

begin
    S := TStringQueue.Create;
    try
        S.Push('Delphi 2007 for Win32');
        S.Push('C++Builder 2007');
        S.Push(' CodeGear RAD Studio 2007');

        writeln(S.Pop);
        writeln(S.Pop);
        writeln(S.Pop);
    finally
        S.Free
    end
end.

```

Met als resultaat de output die we zouden verwachten van een Queue:

```

Delphi 2007 for Win32
C++Builder 2007
CodeGear RAD Studio 2007

```

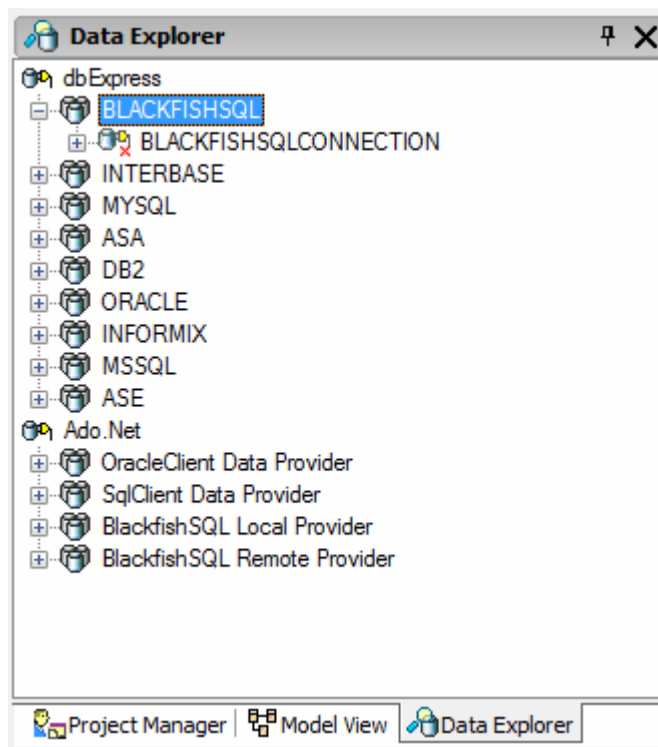
Overgens worden Generics alleen ondersteund in Delphi for .NET 2.0, en niet in de Win32 versie van Delphi.

Volgens de Delphi Roadmap (zie <http://dn.codegear.com/article/36620>) komt ondersteuning voor Generics in Delphi for Win32 pas in de volgende versie van Delphi, met de codenaam Delphi Tiburón (Delphi en VCL development met Unicode en Generics - *wellicht een reden om toch maar een Subscription te nemen?*)

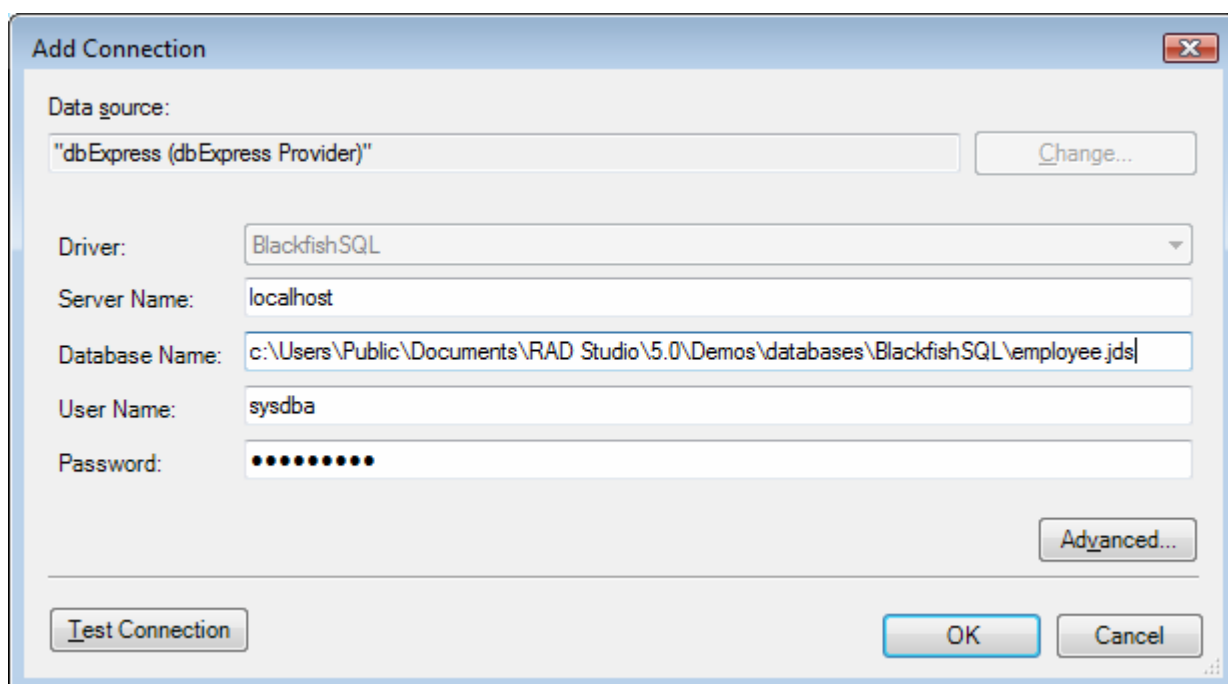
DBX4 en Blackfish SQL

Zoals beloofd is de DBX4 functionaliteit in de CodeGear RAD Studio 2007 uitgebreid ten opzichte van Delphi 2007 for Win32. Maar de uitbreidingen in de Win32 functionaliteit zullen met Update #3 ook in Delphi 2007 for Win32 (en C++Builder 2007) terugkomen. Hierbij dus vast een voorproefje in de vorm van enkele screenshots met kort commentaar.

De Data Explorer in CodeGear RAD Studio is weer uitgebreid. Waar Delphi 2006 nog zowel dbExpress als BDP (Borland Data Provider for .NET) liet zien, en Delphi 2007 for Win32 (en C++Builder) het bij dbExpress hielden, daar laat de CodeGear RAD Studio weel dbExpress en Ado.Net zien:



Merk op dat dbExpress is uitgebreid met een entry voor BlackfishSQL. Dit is een managed SQL database, volledig in Delphi geschreven, die bij de CodeGear RAD Studio wordt meegeleverd.



Blackfish SQL

Blackfish SQL is een managed SQL-compliant embedded database die naast onze Delphi toepassingen kan worden gedeployed. Er zijn drie Blackfish SQL for Windows connection drivers: als eerste DBXClient – een Win32 DBX4 driver om via Delphi en C++Builder naar een remote Blackfish SQL database te praten, als tweede een Local ADO.NET 2.0 Provider (voor een local Blackfish SQL database – de driver en de Blackfish database kernel draaien hierbij in-process), en als derde een Remote ADO.NET 2.0 Provider, waarbij je een .NET remote connection naar een Blackfish SQL database kan maken. Je kan bijvoorbeeld Blackfish SQL als .NET assembly op een web server zetten en er dan ASP.NET toepassingen tegen aan laten werken.

CodeGear RAD Studio 2007 Professional bevat een unlimited Blackfish SQL desktop database deployment license voor systemen met 1 CPU, 1 gebruiker of 4 connecties, met databases die 512 MB kunnen worden.

CodeGear RAD Studio 2007 Enterprise en Architect geven een ongelimiteerde Blackfish SQL database deployment license voor systemen met 1 CPU, maximaal 5 gebruikers of 20 connecties en databases die 2 GB groot kunnen worden.

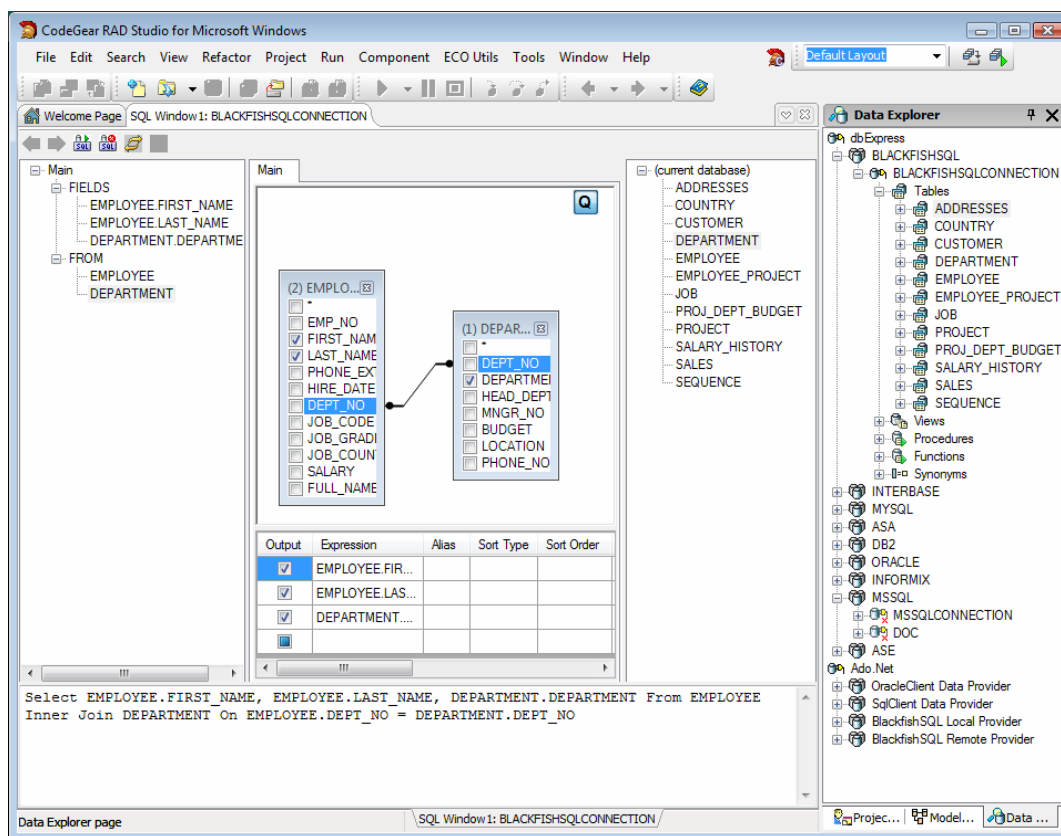
In alle gevallen kunnen extra deployment licenties gekocht worden (ook voor meer gebruikers, connecties, CPUs of grotere databases)– neem contact met me op voor de details als je interesse hebt.

BlackfishSQL gebruik als databases het formaat .jds – hetzelfde als JDataStore, en BlackfishSQL is dan ook de uiteindelijke versie wat destijds als NDataStore begonnen is. Blackfish SQL draait dan ook als managed database onder zowel Java als .NET.

Zie <http://dn.codegear.com/blackfish> voor meer informatie over Blackfish SQL.

Active Query Builder

Een leuke uitbreiding is de Active Query Builder. Hiermee kun je tabellen naar een designer slepen, velden aanklikken die je hebben wilt als resultaat van je query, en vervolgens velden bij elkaar slepen die de tabellen aan elkaar moeten JOINen. Ziet er gelikt uit, en werkt erg makkelijk.



Active Query Builder is ook los verkrijgbaar – zie <http://www.activequerybuilder.com> voor details.

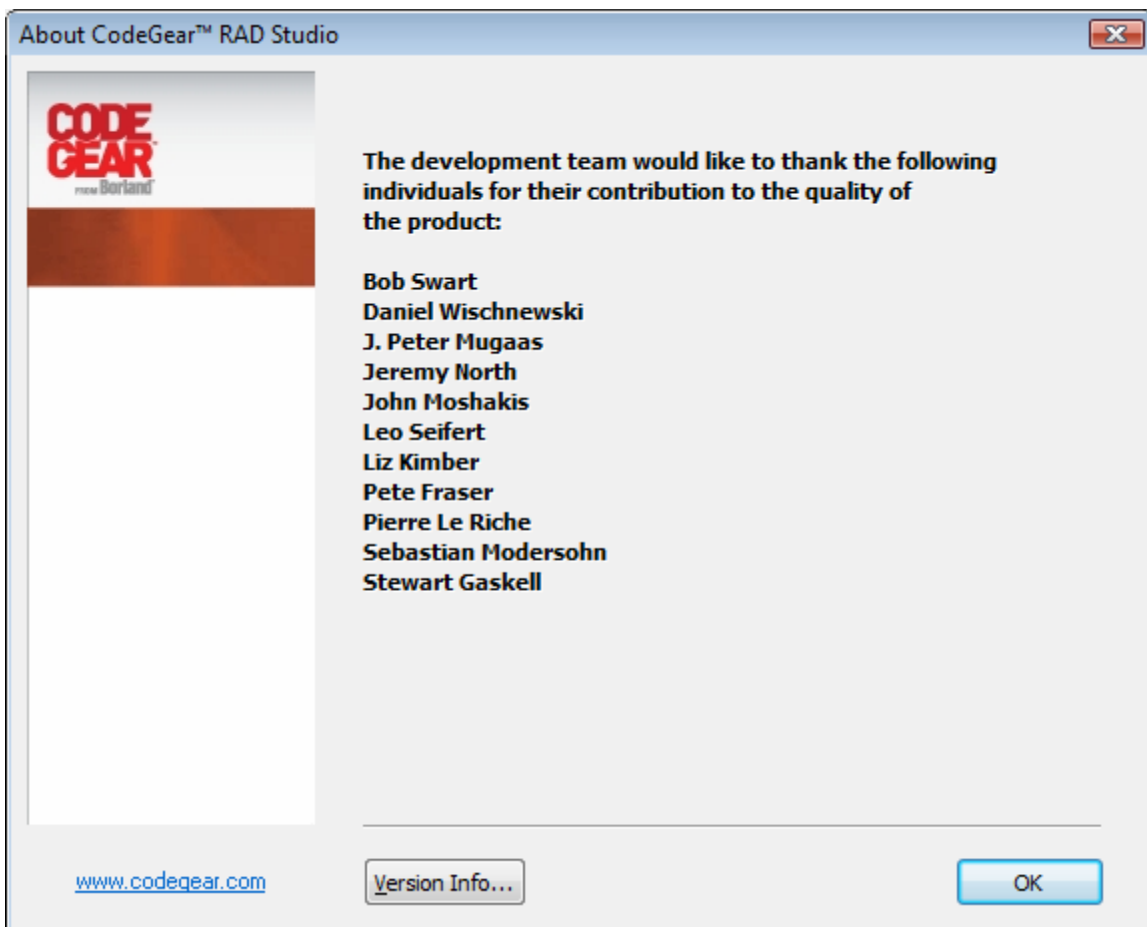
ECO IV

Nu Delphi for .NET 2.0 niet langer WinForms ondersteunt, zat er voor het ECO Team – inmiddels als Capable Objects bezig – niets anders op dan ondersteuning voor de VCL for .NET 2.0 te realiseren. Dat is dan ook meteen een van de grote nieuwe features van ECO IV. Naast ASP.NET 2.0 uiteraard.

ECO IV is alleen beschikbaar in de Architect editie van CodeGear RAD Studio 2007, alhoewel er later ook limited versies van ECO IV voor de overige edities van CodeGear RAD Studio 2007 kunnen komen. Net als IntraWeb (of “VCL for the Web”), is het nu echter een third-party product, met zijn eigen agenda en doelstellingen.

About...

Meer uitbreidingen en features zal ik bespreken zodra de definitieve versie beschikbaar is. Tot slot nog de About box, waar weer een aantal speciale effecten ingebouwd zijn. Alt+KUDOS bijvoorbeeld:





Prijzen

CodeGear RAD Studio 2007 is beschikbaar in een Architect, Enterprise en Professional editie. Onderstaande prijzen zijn exclusief BTW en gelden voor de Electronic Software Delivery (ESD) edities, die je kunt downloaden en kunt installeren met een bijbehorend serienummer dat je ontvangt.

CodeGear RAD Studio 2007 Architect

http://www.codegear.com/article/36898/images/36898/radstudio_architect_rtbuy_us_070817.pdf

| | |
|----------------------------------|------------------|
| New User | € 2.614,= |
| Upgrade | € 1.974,= |
| Subscription CodeGear RAD Studio | € 865,= per jaar |

CodeGear RAD Studio 2007 Enterprise

http://www.codegear.com/article/36898/images/36898/radstudio_enterprise_rtbuy_us_070817.pdf

| | |
|----------------------------------|------------------|
| New User | € 1.974,= |
| Upgrade | € 1.334,= |
| Subscription CodeGear RAD Studio | € 650,= per jaar |

CodeGear RAD Studio 2007 Professional

http://www.codegear.com/article/36898/images/36898/radstudio_professional_rtbuy_us_070817.pdf

| | |
|----------------------------------|------------------|
| New User | € 934 |
| Upgrade | € 414 |
| Subscription CodeGear RAD Studio | € 315,= per jaar |

Special Aanbieding: Wie een CodeGear ontwikkelomgeving bij mij aanschaft in combinatie met de Subscription, krijgt bij mij (gedurende de looptijd van de Subscription) 50% korting op de aanschaf van Delphi cursusboeken in PDF formaat die ik aanbied van <http://www.eBob42.com/courseware> (nieuwe Delphi 2007 boeken komen later dit jaar, en zijn dan ook met korting beschikbaar).

Wie de Enterprise of Architect editie aanschaft kan de korting op cursusboeken ook "inruilen" voor een eenmalige korting van 33% op een Workshop in Helmond (of 25% korting op een Custom Clinic bij u op locatie).

Stuur e-mail naar Bob@eBob42.com voor meer informatie of om een bestelling te plaatsen.

Delphi boeken op Lulu.com

De afgelopen tijd heb ik enkele Delphi cursusboeken omgezet tot een PDF formaat dat bruikbaar was om er door Lulu een boekje van te laten maken. Ik heb twee verschillende formaten aangemaakt: die van maximaal 88 pagina's met een nietje in de rug (de saddle-stitch versie) en de versie vanaf 68 pagina's (maximaal 680) met een zijkaft waarop de titel gedrukt kan worden.

De volgende Delphi titels zijn nu te koop:

- ✍ Delphi 2006 Development Essentials (238 pagina's)
- ✍ Delphi 2006 ADO.NET 1.1 Database Development (169 pagina's)
- ✍ Delphi 2006 Advanced ASP.NET 1.1 Web Development (135 pagina's)
- ✍ Delphi Win32 / .NET Component Development (175 pagina's)

En een paar dunne boekjes, voor 17 Euro per stuk:

- ✍ Delphi for Win32 VCL Database Development (88 pagina's)
- ✍ Delphi 2006 ASP.NET ECO III Weblog (63 pagina's)

Ik werk inmiddels aan een aantal Delphi 2007 titels, waaronder eentje over ASP.NET 2.0 Web Development.

Zie <http://stores.lulu.com/drBob42> voor een compleet overzicht van deze en de andere publicaties die ik op Lulu.com heb gezet, of <http://www.eBob42.com/courseware> voor de cursusboeken die nog niet op Lulu.com staan omdat ik er nog regelmatig training mee geef en uitbreidingen op aanbreng.

Delphi support per e-mail

Naast het geven van training & consultancy (en het spreken op seminars en schrijven van artikelen), ben ik per 1 juni ook begonnen met het geven van structurele ondersteuning per e-mail. Dit betekent dat Delphi ontwikkelaars mij per e-mail vragen en problemen kunnen voorleggen. De prijs zal 100 Euro per (kalender)maand zijn, met een korting voor wie meteen voor een kalenderjaar betaald; dan is de prijs 1000 Euro, ex. BTW. Het eerste kalenderjaar waarvoor dit geldt zal 2008 zijn overigens.

Voor die prijs kan een ontwikkelaar mij vragen per e-mail stellen die ik dan "snel" en in detail zal proberen te beantwoorden. Meestal zal "snel" betekenen binnen een paar uur, vrijwel altijd nog dezelfde dag (er zijn uitzonderingen, als ik bijvoorbeeld die dag training geef of in het buitenland ben). Ik kan helaas geen garantie geven dat ik alle vragen zal kunnen beantwoorden, maar zal wel altijd proberen om relevante informatie of verwijzingen op te zoeken. Mocht de "service" niet bevallen, dan kan er altijd aan het eind van een periode gestopt worden (maar er vind geen restitutie plaats van betaalde bedragen).

Indien een vraag extra uitzoekwerk zou verlangen, dan meld ik dat eerst (plus een schatting hoeveel tijd (en geld) dat extra uitzoeken met zich mee zou brengen). Je kan dan zelf bepalen of het de moeite waard is om mij dat uit te laten zoeken. Iemand heeft mij bijvoorbeeld wel eens laten uitzoeken hoe je een bepaald "vingerafdruk"-scan apparaat kunt gebruiken, waarbij alleen een DLL en een C header file beschikbaar was. Dat is in de vorm van een klein project van enkele dagen uitgevoerd. Tot volle tevredenheid van beide partijen overigens.

Voor de resterende 4 maanden van 2007 biedt ik de ondersteuning aan voor de introductieprijs van 250 Euro (ex.BTW). Deze eenbieding geldt tot eind september 2007. Mail Bob@eBob42.com voor de details.