

# Delphi OplossingsCourant

Vol 4. No. 2.

Een gratis kwartaalpublicatie van Bob Swart Training & Consultancy (eBob42) - <http://www.eBob42.com>



Helmond, 27 mei 2002,

Dit nummer van de Delphi OplossingsCourant staat in het teken van de Borland Conference 2002 vorige week in Anaheim, CA (USA). Het thema van de RAD Product Address was Freedom Today en Freedom Tomorrow, en ik moet zeggen dat me dat zeer aansprak. Niet alleen vanwege het feit dat Borland ons met Java en RAD (onder Linux, Windows en straks .NET) de vrijheid biedt om te kiezen wat we willen, maar ook omdat dit de eerste BorCon is die ik bezoek "als eigen baas". In 1993 bezocht ik mijn eerste BorCon, en vanaf 1994 ben ik ieder jaar als spreker aanwezig geweest, maar dit was de eerste keer dat ik alleen mezelf vertegenwoordigde, en dat gaf me toch ook wel een heel bijzonder gevoel van "Freedom".

Vanaf pagina 8 is mijn BorCon verslag te lezen. De belangrijkste Delphi nieuwtjes betreffen het feit dat Delphi 7 nog dit jaar beschikbaar zal komen, en een preview versie van Delphi voor .NET zal bevatten (bestaande uit de DCCIL command-line compiler en de VCL for .NET FrameWork). De "echte" Delphi voor .NET zal pas later in 2003 beschikbaar komen, maar in ieder geval hebben we al vanaf de tweede helft van dit jaar de kans om echte managed en safe Delphi code voor .NET te gebruiken. Daarnaast komt Kylix 3 nog dit jaar beschikbaar, met zowel een Delphi als C++ IDE.

Behalve BorCon nieuws sta ik in dit nummer van de Delphi OplossingsCourant ook even stil bij het feit dat de BDE officieel bevroren is, en SQL Links zelfs "deprecated" is. Het aanbevolen alternatief voor SQL Links heet dbExpress, en wordt vanaf pagina 3 nader beschreven.

Wie na lezen van de Delphi OplossingsCourant vragen of opmerkingen heeft, kan me gerust een mailtje sturen - ik stel alle feedback op prijs (en dat geldt ook voor "verzoeknummers" betreffende artikelen, columns en/of Delphi trainingen).

## Inhoudsopgave

<b>Welkom, Clinics &amp; Conferenties</b>	<b>1</b>
<b>Delphi 6 Update 2</b>	<b>2</b>
<b>WebSnap Custom Adapters</b>	<b>2</b>
<b>SQL Links "killer" dbExpress</b>	<b>3</b>
<b>Delphi/Kylix Database Development</b>	<b>5</b>
<b>Interview met Jason Vokes</b>	<b>6</b>
<b>BorCon 2002 Nieuws</b>	<b>8</b>
<b>Personal Delphi Agents (PDA)</b>	<b>10</b>
<b>Dr.Bob's Delphi Clinics in 2002</b>	<b>11</b>

De Delphi OplossingsCourant (DOC) is een gratis productie van Bob Swart Training & Consultancy (eBob42).

Eindredactie: Bob Swart

e-mail: [doc@eBob42.com](mailto:doc@eBob42.com)

Het volgende nummer van de Delphi OplossingsCourant zal in augustus 2002 verschijnen, met meer aandacht voor Delphi 7 en Delphi voor .NET.



## Delphi Clinics in 2002

In de 2e helft van 2002 zal er eens in de 2 weken (op donderdag) een Delphi 6 Clinic plaatsvinden in Eindhoven. Data en onderwerpen zijn als volgt:

- 12 sept - **Delphi COM Development**
- 26 sept - **dbExpress & DataSnap**
- 10 okt - **WebBroker & InternetExpress**
- 24 okt - **WebSnap Development**
- 7 nov - **BizSnap: XML/SOAP & Web Services**
- 21 nov - **WebSnap & BizSnap** (in praktijk)
- 5 dec - **.NET FrameWork Development**

voor meer info: <http://www.eBob42.com/training>

## Conferenties 2002

Behalve de Delphi Clinics die begin september weer losbarsten, zal ik ook nog als spreker bij de volgende evenementen aanwezig zijn:

- 9-11 juni - **DCon 2002**, Reading (UK)
- 22-27 sept - **Entwickler 2002**, Frankfurt (D)
- 28-29 okt - **BorCon UK 2002**, London (UK)

## Delphi 6 Update 2

Eind februari 2002 kwam de tweede update voor Delphi 6 beschikbaar (en behalve een hoop verbeteringen is deze update ook noodzakelijk voor wie Delphi 6 naast C++Builder 6 op hetzelfde systeem wil hebben draaien). Bovendien worden er vele zaken verbeterd binnen WebSnap en BizSnap (je zou zelfs kunnen zeggen dat het SOAP deel voor het grootste deel herschreven is). Voor wie altijd het-nieuwste-van-het-nieuwste wil uitproberen is er dus weinig keuze: Delphi 6 Update 2 moet gewoon!

De update kan gedownload worden via de Delphi 6 Registered Users web pagina. Vergeet echter niet om ook de helpfile updates op te halen, want die zaten er nog niet bij! En daarnaast zijn er nog enkele andere zaken die wat bijzondere aandacht vereisen, zoals een nieuwe versie van de Warnings Expert (die het na Update 2 niet meer deed), en wat problemen met Variants op oude(re) versies van Windows...

### Variants

Vlak na het uitkomen van de Update 2 konden we berichten lezen in de delphi.non-technical newsgroep betreffende een probleem met executables (gecompileerd met Delphi 6 Update 2) die niet langer werken onder Windows 95. Gelukkig heb ik thuis nog een Windows 95 machine (een spelletjescomputer voor de kinderen), dus kon ik dit makkelijk testen. Wie schetst mijn verbazing toen ik inderdaad een dialoog kreeg met de melding "Entry Point Not Found" (voor VarNot).

Later bleek zelfs dat dit niet alleen fout gaat op Windows 95 machines, maar ook op de eerste releases van Windows 98 en zelfs Windows NT4 pre-SP2. Het probleem wordt veroorzaakt door het gebruik van een tweetal APIs uit de OleAut32.dll, namelijk VarNeg en VarNot. En laten er nou ruim 20 versies van deze DLL bestaan, die niet allemaal de VarNeg en VarNot bevatten (alhoewel de Microsoft documentatie daar wat vaag over doet).

Om dit probleem "officieel" op te lossen, heeft Borland een nieuwe versie van de Variants, VarUtils en SysConst units uitgebracht, onder de noemer van een Public Beta. Na verloop van tijd zal deze "public beta" de status "officieel" krijgen, en dan kunnen we misschien nog een Update 3 verwachten (het wordt wel weer tijd).

## WebSnap Custom Adapters

Eind vorig jaar deed ik bij een Conference to the Point een sessie over het bouwen (en gebruiken) van WebSnap Custom Adapters bij de Conference to the Point. Helaas was het ruime uur nog steeds aan de korte kant om zowel "normale" WebSnap Adapters als de bouw van Custom Adapters goed uit te leggen, waardoor ik denk dat niet iedereen deze sessie evengoed heeft kunnen volgen. Ter verduidelijking heb ik toen een artikel geschreven over WebSnap Adapters, dat goed is ontvangen.

Echter, in de tussentijd heb ik niet stilgezeten, en om een-en-ander nog eenvoudiger te maken (of dat in ieder geval te proberen) heb ik een kleine toepassing geschreven die in staat is om - op basis van de namen van de Adapter Fields en de naam van de Adapter zelf - de complete source code voor WebSnap Custom Adapters te genereren! Dit scheelt een hoop werk, want een nieuwe WebSnap Custom Adapter met zo'n 10 Adapter Fields levert al snel rond de 1000 regels source code op (en die wil ik niet graag met de hand intikken).

Op <http://www.eBob42.com/columns> is de april 2002 column te vinden met de internet-versie van de WebSnap Custom Adapter Generator (er komt binnenkort ook een Win32 GUI versie beschikbaar met enkele extra opties, die ik voor het eerst presenteerde tijdens een *Birds-of-a-Feather* sessie van BorCon 2002 in Anaheim). De gegenereerde source code is overigens compatible met Delphi 6, C++Builder 6 en Kylix 2.

Elk Adapter Field krijgt een automatische check om te zien of er wel iets ingevuld is. De gebruiker moet daar straks dus iets invullen. Dit kun je aanpassen door het OnValidate event voor het specifieke Adapter Field in te vullen (de WebSnap Custom Adapter krijgt voor ieder Adapter Field een OnValidateXXX waarbij XXX de naam van het Adapter Field is). Door in deze event handler het Handled argument op True te zetten wordt de oorspronkelijke OnValidate niet langer aange-roepen. Ook is het mogelijk om hierbinnen zelf je eigen checks te doen (zoals het controleren van een postcode bij een NAW\_Adapter).

Naast de automatische checks, zullen de Adapter Fields hun waarde automatisch behouden. Mocht dus de gebruiker een foute postcode invullen (of een leeg e-mail adres), dan zullen de overige Adapter Fields hun oude waarde houden (en niet leeg worden, zoals bij de LoginFormAdapter die bij een foute login alle velden leeg maakt).

## SQL Links *killer* dbExpress

*Enkele dagen voor de Borland Conference in Anaheim verscheen een artikel op de Borland Community website met daarin de toekomst beschreven van de BDE en SQL Links. Om kort te gaan: de BDE is bevroren, en SQL Links "depricated" (aanbevolen alternatief: dbExpress).*

### DataSnap Direct

Hetzelfde artikel gaf aan dat dbExpress vanaf volgende versie(s) van Delphi, C++Builder en Kylix de naam DataSnap Direct zal krijgen. Maar omdat die nieuwe versies nog niet beschikbaar zijn, zal ik hier gewoon gebruik blijven maken van de "oude" naam dbExpress.

We zagen dbExpress voor het eerst in Kylix, toen een cross-platform data access benadering nodig was, die zowel onder Linux als onder Windows zou kunnen werken. Al vanaf het eerste begin is dbExpress opgezet om te zijner tijd de SQL Links te vervangen, en die tijd is dan nu gekomen.

Voor wie nog niet eerder de stap heeft genomen om dbExpress te gebruiken (of te bekijken), zal ik nu eerst kort de componenten beschreven die onderdeel uitmaken van dbExpress, en daarna laten zien hoe deze componenten in praktijk samenwerken (en waar je speciaal op moet letten).

### dbExpress

De dbExpress tab van Delphi 6 bevat een zevental componenten. Van links naar rechts hebben we de TSQLConnection, TSQLDataSet, TSQLQuery, TSQLStoredProc, TSQLTable, TSQLMonitor en TSQLClientDataSet.

### TSQLConnection

De TSQLConnection component is de verbinding tussen de dbExpress driver en de (DataCLX) componenten van de dbExpress tab. De eerste property die we een waarde moeten geven is de ConnectionName. Afhankelijk van de versie van Delphi (of Kylix) kunnen we hier kiezen uit connecties voor o.a. InterBase, MySQL en Oracle. Een keuze voor ConnectionName bepaalt tevens de waarde van de VendorLib en de (dbExpress) LibraryName properties. De VendorLib is de DLL of .so file die door het DBMS wordt meegeleverd.

Door te dubbelklikken op de TSQLConnection component krijgen we de Connection Settings dialoog te zien, waarin we zelf de verschillende parameters voor de gekozen ConnectionName kunnen invullen (zoals de plaats van de database, en de username/password).

Met de Connected property kunnen we dan een daadwerkelijke verbinding maken met de database (maar nog geen data ophalen).

### TSQLDataSet

Als we een actieve TSQLConnection hebben, kunnen we gebruik maken van een TSQLDataSet om daadwerkelijk data op te halen. De TSQLDataSet is hier de meest flexibele component voor, omdat deze zich - via de CommandType property - kan voordoen als Query, Tabel of Stored Procedure. Afhankelijk van deze keuze, zal de waarde van de CommandText property op een bepaalde manier worden geïnterpreteerd (en is het resultaat equivalent aan een TSQLQuery, TSQLTable of TSQLStoredProc).

### TSQLQuery

De TSQLQuery praat via de TSQLConnection property met een TSQLConnection component, en kan een SQL Query kwijt in zijn SQL property. Het resultaat is wederom een unidirectionele read-only dataset (net als bij de TSQLDataSet).

### TSQLStoredProc

De TSQLStoredProc praat met een TSQLConnection component en kan daarmee een stored procedure uitvoeren.

### TSQLTable

De TSQLTable praat met een TSQLConnection component en kan een hele tabel opvragen. Wederom unidirectioneel en read-only.

### TSQLMonitor

De TSQLMonitor component kan gebruikt worden om de SQL statements (select, update) die naar de TSQLConnection component worden gestuurd te laten zien via de TraceList property.

### TSQLClientDataSet

Het gebruik van een TSQLDataSet (of TSQLQuery, TSQLTable of TSQLStoredProc) resulteert in een unidirectionele read-only dataset. Om de data hiervan te gebruiken in combinatie met data-aware controls moeten we deze data via een DataSetProvider in een ClientDataSet component stoppen (in feite een in-memory cache met de opgehaalde records). Wie geen zin heeft om deze twee extra componenten te gebruiken, kan ook de speciale TSQLClientDataSet component gebruiken die in feite een combinatie is van de TSQLDataSet, TDataSetProvider en TClientDataSet. Eigenlijk alleen geschikt voor eenvoudige(re) toepassingen.

## dbExpress in Praktijk

Het gebruik van de dbExpress componenten gaat iets anders dan de BDE componenten die we gewend zijn te gebruiken - met name door het feit dat we nu met unidirectional read-only datasets werken in dbExpress.

We beginnen met de TSQLConnection component en kiezen een waarde voor de ConnectionName property, zoals IBLocal. Klik met de rechter muisknop op de SQLConnection component om de IBLocal Connection Settings te veranderen (en o.a. de Database naar een juiste plek te laten wijzen, en als prefix de naam van de machine te geven, zoals de `voyager:d:\delphi6\data\employee.gdb`). Als we ook de User\_Name en Password hebben ingevuld kunnen we de LoginPrompt property op False zetten - is wel zo prettig in het gebruik.

Zet nu een TSQLDataSet component naast de TSQLConnection, en gebruik de SQLConnection property om ze aan elkaar te verbinden. De waarde van CommandType staat default op ctQuery, zodat we in CommandText een SQL query kunnen invoeren. Klik op de drie puntjes om de SQL Query Designer te starten. Maar je kan ook de CommandType op ctTable zetten en dan in de drop-down combobox van CommandText een tabelnaam kiezen. Wat je ook kiest, het resultaat zal een read-only dataset zijn die maar in één richting te bewandelen is (van begin tot eind). Dit kan nuttig zijn in web server toepassingen, waar je maar één keer door je resultaat heen hoeft te lopen en de record naar HTML kan om (laten) zetten. Maar voor visuele toepassing die gebruik maken van data-aware controls, en niet alleen heen maar ook terug moeten kunnen navigeren, zullen we iets extra's moeten doen.

Zet daartoe zowel een TDataSetProvider als een TClientDataSet component naast de TSQLDataSet component. Laat de DataSet property van de TDataSetProvider wijzen naar de TSQLDataSet component, en de ProviderName property van de TClientDataSet naar de TDataSetProvider. Als we nu de Active property van de TClientDataSet op True zetten, zal deze alle records opvragen (via de TDataSetProvider, uit de TSQLDataSet).

Met behulp van een TDataSource kunnen we tot slot de inhoud van de TClientDataSet in een TDBGrid laten zien. Een TDBNavigator zorgt ervoor dat we ook probleemloos heen en weer kunnen lopen door de dataset. Dit is niet mogelijk zonder de TDataSetProvider en TClientDataSet.

## dbExpress Updates

Maar er komt nog meer bij kijken. Door gebruik te maken van een DataSetProvider en ClientDataSet hebben we de read-only unidirectional dataset in feite in een lokale buffer gestopt (de ClientDataSet). Maar wijzigingen worden dan ook alleen lokaal gemaakt, en niet automatisch doorgestuurd naar de onderliggende SQLDataSet of zelfs het DBMS. Hiervoor moeten we expliciet de ApplyUpdates methode aanroepen, met als argument het aantal "update errors" dat we willen toestaan voor we verdere updates (in dezelfde aanroep) willen afbreken. Een aanbevolen waarde voor dit argument is 0, zodat er niet verder wordt gegaan met updates zodra er een record aangetroffen wordt dat - in de tussentijd - al door een andere gebruiker is gewijzigd. We kunnen zelf op deze situatie reageren in het OnReconcileError event.

### ApplyUpdates

De grote vraag is echter: *hoe en waar is de beste plek (en wanneer is het beste tijdstip) om de aanroep van ApplyUpdates te doen?* Nog even voor de duidelijkheid: zonder ApplyUpdates zullen alle wijzigingen in de ClientDataSet weer verloren gaan als de toepassing wordt afgesloten - het is immers slechts een dataset in het geheugen.

Aan de andere kant voert het misschien wat ver om aan de eindgebruiker zelf te vragen om (regelmatig) op de zgn. "ApplyUpdates" knop te drukken. Een betere oplossing is om bij het afsluiten van de toepassing een check in te bouwen die kan zien of er nog updates zijn die geapplied moeten worden, met de volgende code:

```
if ClientDataSet.ChangeCount > 0 then  
    ClientDataSet.ApplyUpdates(0)
```

Eventueel kun je hier nog een dialoog plaatsen om te vragen of de wijzigingen wel opgeslagen moeten worden.

### OnAfterPost en OnAfterDelete

Een andere (extra) mogelijkheid is om de aanroep van ApplyUpdates zowel in de OnAfterPost en in de OnAfterDelete event handlers te plaatsen (merk op dat de on-line help alleen de OnAfterPost event handler aanraadt, en vergeet dat er dan niet op een delete wordt gereageerd, zodat verwijderde records pas echt worden verwijderd als er nog een latere OnAfterPost volgt). Voor meer details verwijst ik graag naar mijn dbExpress en DataSnap Clinic op 26 september in Eindhoven.

# Book Review: Delphi/Kylix Database Development

Delphi/Kylix Database Development is a book of 400 pages about a relative narrow topic. Which is a good thing, as it saves you from having to skip the first few hundred pages that introduce Delphi and/or ObjectPascal (as I often encounter in Delphi books). This book assumes you already know Delphi and ObjectPascal, but have little or no experience using dbExpress - the new cross-platform data-access layer for Delphi, Kylix or C++Builder.

The book consists of nine chapters and two appendices. Before the chapters start, a small introduction explains what the book is all about (mainly dbExpress), and what to expect from it.

The first chapter, "Establishing and Using Database Connections", starts with dbExpress and shows how to make a connection to a database using dbExpress (often the starting point indeed).

Chapter 2, "dbExpress Datasets", builds on this example and gives an overview of the dataset components found in dbExpress, showing how to use them and when to use which one.

Chapter 3, "Client Dataset Basics", covers the TClientDataSet component, which plays an important role in both dbExpress and DataSnap (a topic from a later chapter).

Chapter 4, "Advanced Client Dataset Operations" continues the coverage of the ClientDataSet, discussing some advanced features such as nested datasets, BLOBs, undo support, etc. No reconcile error handling, yet, because that will be covered in the DataSet Provider chapter.

Chapter 5, "Data-Aware Components" is a bit of a basic chapter for more Delphi developers, since we have probably worked with data-aware components for years now. The best thing about the chapter is that it also shows how to write your own custom data-aware components, building a sample data-aware TDateTimePicker.

Chapter 6, "Data-Aware Grids" covers TDBGrid but goes in-depth when it comes to the things you can do with it (I actually learned some new techniques). It also covers TClientDataSetGrid, written by John Kaster, and at the end of the chapter lists a few third-party grid components.

Chapter 7, "DataSet Providers" covers the TDataSetProvider component, which is the

foundation for the multi-tier capabilities of Delphi and Kylix. This chapter contains lots of details, and also covers the reconcile error dialog. Any DataSnap developer should read this chapter, and will probably learn a lot from it.

Chapter 8, "DataSnap" is mainly for Delphi developers, since it covers multi-tier applications using Delphi-specific connection components.

Chapter 9, "The ConMan Application" covers a contact management Delphi application as an example multi-tier database application in the real-world.

With both Delphi and Kylix in the title, *just how much of the book is Delphi specific, and how much is Kylix specific?* Well, it turns out that the book is about Delphi using mainly the cross-platform CLX (which is supported by both Delphi and Kylix). Sometimes a reference is made to VCL-specific issues, and parts of the DataSnap chapter are also Delphi only. Apart from that, all of the 50+ screenshots are made on Windows, so the book feels mainly as a Delphi Database Development book, with the added (automatic) benefit that dbExpress in itself is cross-platform. You should think twice, however, if you're a pure Kylix developer, since there may be Kylix/Linux specific issues that are not addressed in this book.

All in all, it's a good book with lots of detail when it comes to dbExpress (and even DataSnap - formerly known as MIDAS - gets a more than decent coverage). The fact that the author was assisted by the Borland engineers responsible for dbExpress and with intimate knowledge of DataSnap really helped to get the book to this level. The only disadvantage that I can think of is that dbExpress isn't static, and new drivers (and add-on components) are still being developed - something not addressed in this book, so you may want to look for additional resources such as the Borland Community website for up-to-date driver information. But regardless of that, I often mention the book at my Delphi 6 Clinics on dbExpress or DataSnap, and in fact I can highly recommend it to any Delphi dbExpress database developer.

**Author:** Eric Harmon (with Dan Miser, Ramesh Theivendran and Phillippe Bruno)

**Publisher:** SAMS

**ISBN:** 0-672-32265-X

**Price:** US\$ 49.99

**Info:** 414 pages, no CD-ROM

# Interview Jason Vokes

De Conference to the Max werd gehouden in de week voor BorCon 2002. Tegelijk met mijn Active Server Objects in Delphi sessie heeft Jason Vokes van Borland EMEA (Europe, Middle-East, Africa) een belangrijke sessie gedaan over The Future of Delphi and Kylix: *"Attendees of this session will learn more of the Borland strategy for providing the RAD development tools that will be needed by developers to face new challenges on both the Windows/.NET and Linux platforms."*

Na afloop sprak ik met Jason Vokes en mocht ik een kort interview met hem houden, waarvan hier het verslag. Dit interview is een leuke inleiding voor het BorCon 2002 nieuws dat op pagina 8 begint.

## Crossplatform

In het interview met Jason Vokes lag de nadruk op crossplatform RAD tools (dus Delphi, C++Builder en Kylix - JBuilder werd buiten beschouwing gelaten). Linux, Windows en .NET werden in één adem genoemd. Belangrijk hierbij is de source code: die zou zo min mogelijk aangepast moeten worden, zodat je bijna met single source op alle platforms kunt werken (ook al kan dat een aantal IFDEFs betekenen).

De huidige stand van zaken betreft met name de ondersteuning voor WebServices: vanuit Delphi 6, Kylix 2 en C++Builder 6 zijn op dit moment WebServices te maken die te gebruiken zijn in een .NET omgeving, en andersom (we kunnen in C# WebServices bouwen die vanuit Delphi te gebruiken zijn). Maar dat is niet genoeg natuurlijk, we willen meer!

## Kylix 3 - Summer 2002

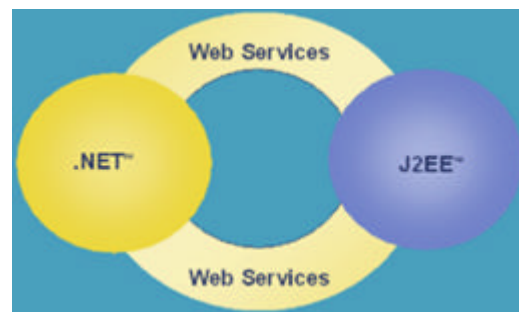
Onder het kopje "Kylix 3" en "zomer 2002" werd me duidelijk gemaakt dat Borland nog steeds serieus met Kylix aan de weg timmert. De volgende versie zal zelfs zowel Delphi als C++Builder voor Linux in één doos bevatten (dus eindelijk de groei naar elkaar toe). Of dit ook iets kan betekenen voor Delphi en C++Builder voor Windows (of later .NET) werd ontwekend beantwoord - de tijd zal het leren. Jason liet vervolgens de Kylix 3 IDE zien, en bouwde een C++ WebService onder Linux. Mijn indruk is dat Kylix 3 - met C++ ondersteuning - door de Linux wereld waarschijnlijk goed ontvangen zal worden, zeker als er ook weer een Open Edition van verschijnt.



Jason Vokes (Borland)

## .NET en J2EE

Alvorens zich helemaal op .NET te richten, liet Jason een slide zien waarop de wereld van .NET en die van J2EE (Java 2 Enterprise Edition) met elkaar verbonden waren door middel van WebServices (voor JBuilder heeft Borland inmiddels de WebServices Kit for Java uitgebracht). Borland maakt het mogelijk om beide frameworks (.NET en J2EE) te laten samenwerken. Voorlopig alleen nog via WebServices, maar er zal meer komen.



## Evolve to .NET

Tijdens de keynote van Ron Tolido (die trouwens wel heel erg tekeer ging tegen Java, en waarschijnlijk "vergeten" was dat in 1999 het TAS-AT team de tweede prijs in de RAD Race won met JBuilder en JDeveloper), gaf deze aan dat er in Visual Studio .NET flink wat veranderingen zijn aangebracht. Met name het Framework is flink gewijzigd (het zou "zeer" zijn om te kiezen voor de C++ of VB gebruikers), om niet te zeggen helemaal nieuw. Dat betekent een behoorlijke investering, in kennis en code (migratie?) voor wie met Visual Studio .NET aan de slag wil.

De bedoeling van Borland's .NET oplossing is om een migratie naar .NET mogelijk te maken, die meer een evolutie is. Net als bij Delphi en Kylix, moet het straks mogelijk zijn om met een minimum aan veranderingen de code klaar te maken voor .NET (althans, dat is de bedoeling).

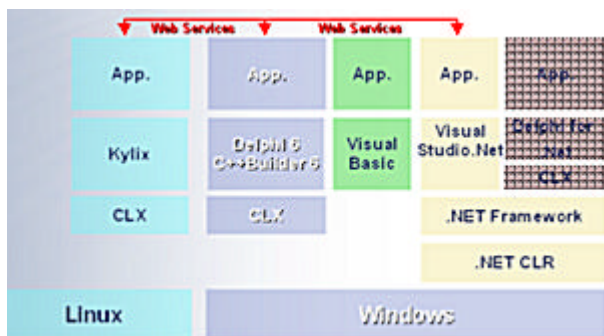
Het doel is uiteindelijk om .NET managed code te genereren. De source code die gebruikt wordt, noemde Jason CLX - en dat deed meteen een belletje rinkelen.

## CLX

CLX staat voor Component Library for X-platform, en werd al eerder gebruikt voor crossplatform ontwikkelingen van Delphi en Kylix (en C++Builder overigens). CLX bestaat uit vier delen: BaseCLX (de system units en low-level zaken), VisualCLX (de visuele componenten, DataCLX (de data access componenten, met o.a. dbExpress en DataSnap als belangrijk onderdeel) en NetCLX (de internet componenten).

*Tijdens BorCon 2002 werd er zelfs over MobileCLX gesproken - drie keer raden wat daar in zal komen.*

Bij implementatie van CLX voor Kylix is VisualCLX gebouwd bovenop de Qt library, omdat daar zowel een Linux als een Windows variant van bestond. Dit wil echter helemaal niet zeggen dat VisualCLX voor .NET ook van Qt gebruik gaat maken. Integendeel: CLX voor .NET (om die term maar even te gebruiken) zal gewoon bovenop .NET en het .NET FrameWork gebouwd zijn. Echter, de componenten die CLX beschikbaar stelt aan de Delphi ontwikkelaar zullen de "bekende" componenten zijn die we al jaren gebruiken: bijvoorbeeld TEdit en geen TextBox.

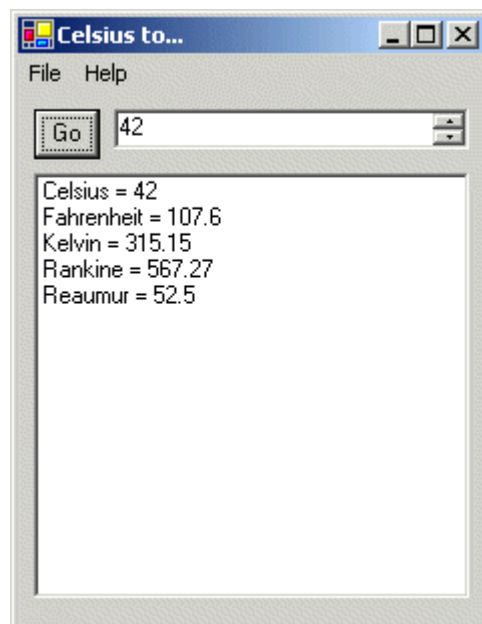


In bovenstaand plaatje zien we de verschillende kolommen voor Kylix met CLX (boven Linux), Delphi 6/C++Builder 6 met CLX, Visual Basic, Visual Studio.NET en helemaal rechts Delphi for .NET (slecht te lezen), die net als Visual Studio.NET bovenop het .NET FrameWork staat, en gebruik maakt van de .NET CLR (Common Language Runtime).

Het is op dit moment al mogelijk om via Web Services te communiceren (en N-tier toepassingen te maken) met applicaties onder Linux, Windows en .NET.

## DCCIL

Om een-en-ander te demonstreren liet Jason vervolgens de Delphi for .NET command-line compiler zien, genaamd DCCIL. Deze kan van .pas bestanden .dcul bestanden maken (dus steeds overal de "il" extra achter). Er was ook een demo project genaamd ConvertIt.dpr, waarvan Jason de source code liet zien. Opvallend was de uses clause, waar namespaces werden gebruikt (zoals de Borland.Strings unit). Verder zag het er redelijk standaard Delphi uit, met uitzondering van een routine ReadState die de definitie van het Form zelf bevat - in Pascal syntax. Er was inderdaad geen .DFM (of .XFM) bestand te zien. Het is nog te vroeg om te zeggen of dit zo blijft, of alleen maar een noodzaak was om de command-line compiler te laten werken. Na compilatie kregen we in ieder geval een ConvertIt.exe (een .NET managed toepassing van amper 7 KB) die een .NET Windows Forms liet zien waarmee we temperaturen konden converteren:



## Onder voorbehoud

Jason maakte me duidelijk dat de demo van de command-line compiler dezelfde demo was die ze in de USA bij TechEd hadden gedaan. En toen hij een "dir" deed, zag ik ook dat de datum van de command-line compiler op 9 april 2002 stond. In de tussentijd zijn ze natuurlijk niet stil blijven staan bij het RAD R&D Team bij Borland, zoals in het BorCon 2002 Nieuws artikel te lezen valt.

Als laatste quote verliet Jason Vokes me met de mededeling dat "the .NET compiler will be productised and made available in the second half of this year."

# BorCon 2002 Nieuws

*Van zaterdag 18 t/m woensdag 22 mei ben ik spreker en bezoeker geweest bij de Borland Conferentie in Anaheim. Onderstaand verslag bevat met name de Delphi en Kylix gerelateerde zaken.*

*Een uitgebreider en dagboek-achtig verslag (in het Engels) is te vinden op <http://www.drBob42.com/BorCon.htm>*

## Kylix 3

Binnenkort zal Kylix 3 uitkomen. De grootste nieuwe feature van Kylix 3 is het feit dat Kylix 3 zowel een Delphi als een C++ IDE zal bevatten (in dezelfde doos). Het is nog onduidelijk wanneer Kylix 3 exact beschikbaar zal zijn - Jason Vokes had het in mijn CttM 2002 interview met hem over "zomer 2002" (en dat duurt niet lang meer). In een volgend nummer van de DOC zal ik hier uiteraard uitgebreid op terug komen.

## Delphi 7

Later in 2002 kunnen we Delphi 7 voor Windows verwachten (op dit moment beter bekend onder de codenaam **Aurora**).

Nieuwe features van Delphi 7 zijn niet bekend gemaakt, of het moet de voorbereiding op .NET zijn. Deze voorbereiding komt in de vorm van speciale nieuwe compiler hints en warnings, en een behoorlijk aantal uitbreidingen in de syntax van de Delphi programmeertaal zelf (daar kom ik hiernaast nog op terug).

Daarnaast zal Delphi 7 een zogenaamd preview versie van Delphi voor .NET bevatten. Deze preview versie bestaat uit de DCCIL command-line compiler alsmede de VCL componenten voor .NET Framework. Op het eerste gezicht lijkt dit niet zoveel, maar er is me verzekerd dat dit voldoende is om daadwerkelijk toepassingen te compileren en te deployen onder .NET (maar dan moeten we wel gebruik maken van de "gewone" Delphi 7 code editor of - voor de liefhebber - visual notepad).

## Delphi for .NET

Pas later in 2003 kunnen we de echte Delphi voor .NET verwachten (merk op dat de naam niet Delphi.NET kan en mag zijn, dus het is Delphi voor .NET - tenzij er nog een andere codenaam uit de hoge hoed komt rollen).

Delphi voor .NET zal beschikbaar komen in een geheel nieuwe Development Studio for .NET waar op dit moment hard aan gewerkt wordt (onder de codenaam **Galileo**).

## Delphi Evolutie

Er werd tijdens de Borland Conferentie in Anaheim veel aandacht besteed aan .NET en de gevolgen die .NET heeft voor (met name) Delphi.

Danny Thorpe (senior R&D engineer van het Borland RAD Team) liet er in zijn sessies geen twijfel over bestaan dat in zijn ogen de stap van Delphi naar .NET evengroot (zoniet groter) zal zijn als de overgang van Borland Pascal naar Delphi. Wie net als ik al zolang meegaat ("je wordt ouder Papa") kan zich vast nog wel herinneren dat er in Borland Pascal van TObject gebruik gemaakt wordt, net als de Object Windows Library (OWL). In Delphi gebruiken we echter TClass en de Visual Component Library (VCL), met daarin tevens het property-method-event model.

*Wat zal er in Delphi voor .NET allemaal anders zijn?* De details zijn nog niet bekend gemaakt, maar enkele sessies in Anaheim hebben al wel een overzicht gegeven van de zaken die we kunnen verwachten, en die ook al in Delphi 7 aanwezig zullen zijn zodat we straks al in Delphi 7 van de nieuwe taalelementen gebruik kunnen maken om de overstap naar Delphi voor .NET straks wat makkelijker te maken dan - bijvoorbeeld - de overstap van Borland Pascal naar Delphi (of van Delphi 5 naar Kylix).

## Delphi 7 Language Features

De lijst met nieuwe taalelementen is lang, en van lang niet alle elementen ligt vast (of is vertoond) hoe die er uiteindelijk uit komt te zien. Wat genoemd is, is in ieder geval de volgende lijst - in redelijk willekeurige volgorde:

### - Unit namespaces

deze bepalen de unieke scope van classes, variabelen, routines, etc. Een beetje te vergelijken met de huidige naam van units. Teneinde namespaces goed te kunnen implementeren zullen units een naam krijgen waarin punten (dots) kunnen voorkomen, teneinde de namespace aan te geven, zoals bijvoorbeeld de units Borland.Delphi.System en eBob42.Euro42.

### - Geneste types en constanten

oftewel type declaraties en const declaraties binnen de scope van een class declaratie zelf.

### - Class static methods

oftewel class methods zonder Self pointer - een beetje vergelijkbaar met class methods zoals we die nu al kennen (die je kunt aanroepen zonder een instantie van de class zelf te hebben).



### - **Class fields en properties**

velden en properties kunnen nu zelf classes zijn.

### - **(super) Records**

met ondersteuning voor inheritance, static methods, properties, nested types, en const declaraties. Oftewel: het verschil tussen een record en een class wordt een stuk kleiner.

### - **Sealed classes**

oftewel een class waarvan niet meer verder af te leiden is. Kan behoorlijk nuttig zijn in sommige situaties.

### - **Final methods**

dit zijn methods die wel virtual of dynamic waren, maar niet verder meer mogen worden verbijzonderd met override. De method versie van de sealed class dus als het ware.

### - **Attributes**

een attributes is een regel (tag) tussen vierkante haken (dat is de syntax in ieder geval) die effect heeft op de daaropvolgende regel code - meestal een declaratie van een class, method, veld, variabele, argument, etc. Een attribuut bestaat uit een naam en optioneel wat argumenten. De tag wordt meegecompileerd, en kan gebruikt worden door zowel de compiler als de run-time library - afhankelijk van het type attribuut.

Er zullen een behoorlijk aantal attributen vooraf gedefinieerd zijn, zoals [WebMethod] of [XML] maar ook zal het mogelijk zijn om zelf eigen attributen te definiëren via een custom class die afgeleid is van TCustomAttribute. De eventuele optionele argumenten zullen ook de argumenten van de constructor van deze class zijn. Zodra ik hier wat meer over weet (voor of vlak na de release van Delphi 7) volgen uiteraard de nodige artikelen over deze nieuwe features.

### - **Qualified identifiers**

Identifiers kunnen straks ook Unicode tekens bevatten, alhoewel Klingon - aldus Danny Thorpe - nog steeds niet ondersteund zal worden.

### - **Multicast events**

Net als bijvoorbeeld in Java, zal het straks mogelijk zijn om je object als "luisteraar" te registreren bij een multicast event.

### - **Array property overloads**

het is nog niet helemaal zeker of dit al in Delphi 7 zal zitten, maar uiteindelijk zal het er wel komen.

## Safe Managed Code

Het belangrijkste feit is overigens dat Delphi voor .NET zich richt op *safe, managed code*. Dus niet unmanaged code (wat we nu al met Delphi 6 kunnen doen), en ook niet unsafe, zodat er enkele zaken in de huidige versie van Delphi zullen zijn die we straks niet meer terug zullen zien...

### Niet in Delphi voor .NET

De volgende zaken in Delphi werden duidelijk gepresenteerd als (potentieel) gevaarlijk, niet of nauwelijks (als safe) onder te brengen in managed code, en zullen derhalve niet in Delphi voor .NET terug te vinden zijn:

- @ en Addr
- Absolute
- Real48 (het 6-byte real type)
- File of Type
- Typeless var parameters
- GetMem/FreeMem/ReAllocMem
- ExitProcs
- "Oude" Turbo/Borland Pascal type TObjects

Het ligt in de lijn der verwachtingen dat deze zaken nog wel in Delphi 7 beschikbaar zijn, maar dan resulteren in een compiler hint of warning, om aan te geven dat deze taalelementen niet aanwezig zijn in Delphi voor .NET.

## DataSnap

DataSnap zal straks in niet minder dan drie verschillende edities beschikbaar komen. Wat we nu kennen als DataSnap (de nieuwe naam voor MIDAS), zal straks **DataSnap Middleware** gaan heten. De huidige naam voor dbExpress zal straks **DataSnap Direct** zijn, en er komt tot slot ook nog - speciaal voor Paradox en dBASE tabellen - een **DataSnap BDE Local**. Een DataSnap ADO (voor dbGO for ADO) zullen we echter niet zien.

## Charlotte

Tot slot nog even aandacht voor het laatste project van Chuck J. genaamd **Charlotte** (volgens Chuck werd het - na Delphi en Kylix - hoog tijd dat ze weer eens een codenaam bedachten die weinig kans had om de naam van het uiteindelijke product te worden). Charlotte is een ontwikkel-omgeving die specifiek gemaakt wordt voor de ondersteuning van Web Services. Het is de bedoeling dat het de *de facto* standaard zal worden om nieuwe Web Services in te definiëren en realiseren.

# Personal Delphi Agents

Wie zich wil certificeren op Delphi gebied heeft een aantal mogelijkheden. Naast de certificering door Borland zijn er ook andere bedrijven die zich hiermee bezighouden, zoals BrainBench en PDA (Personal Delphi Agents).

Ik ben zelf betrokken bij de distributie van de PDA Delphi en Kylix certificatie (dus ik ben niet geheel onpartijdig). Het examen bij PDA bestaat uit een on-line test van 60 (engelstalige) multiple choice vragen, waarbij je per vraag een minuut bedenktijd krijgt. De vragen zijn samengesteld (en getest) door een groot aantal ervaren Delphi en Kylix auteurs en ontwikkelaars.

Het examen kan op vier verschillende niveaus afgenomen worden, namelijk Junior, Intermediate, Senior en Guru. Je moet 70% scoren (42 van de 60 vragen) om het examen te halen, en het certificaat is 3 jaar geldig. Naast het certificaat krijg je de mogelijkheid om je score te bekijken. De top 10% komen in de PDA Delphi Developers Hall of Fame, en de kandidaat met de beste score ontvangt een exemplaar van Delphi 6 Professional.



Op dit moment zijn examens beschikbaar voor Delphi 5, Delphi 6 en Kylix 2. De prijs is \$49.99 - zie <http://www.drBob42.com/PDA> voor meer info.

## Delphi 6 Clinic Quotes

In de eerste helft van 2002 heb ik weer enkele quotes ontvangen van deelnemers aan de Delphi 6 Clinics in Eindhoven. Een korte bloemlezing:

*"Wij volgen Dr.Bob's Delphi Clinics voornamelijk om snel op de hoogte te komen van de mogelijkheden van de nieuwe technieken welke de Borland ontwikkel-gereedschappen geven. Zo besparen we niet alleen veel tijd, maar houden er ook nog gedetailleerde naslagwerken aan over met veel praktisch toepasbare informatie"*

Akkermans & Partners Software B.V.

*"De beste opleiding om mee te zijn met Delphi en internet ontwikkelingen!!"*

General Computers NV (België)

*"Delphi Clinics: Spits-technologie verwerkt in een Blitz-cursus!"*

ACEA (België)

*"De inhoud van de clinics is duidelijk, alsmede de documentatie (lesmateriaal) welke achtergelaten wordt. Wanneer we na een clinic vragen hadden over de besproken stof (over andere Delphi vragen) wordt er zeer snel met een duidelijk antwoord gereageerd, we hebben het wel eens anders gezien!"*

IDB Telematica B.V.

*"Je Delphi Clinics zijn de investering dubbel en dik waard!"*

Plus Plus Software & Services BVBA (België)

*"Wat mij betreft waren de trainingen ZEER leerrijk en interessant. De manier waarop je de training verzorgt kan mijns inziens enkel respect afdwingen voor je parate kennis op dit vlak.*

*Hoewel het tempo soms (aangenaam) hoog lag, ben je er wat mij betreft toch telkens in geslaagd de dag op een onderhoudende manier door te komen, zonder het zo bekende (en gevreesde) slaapeffect na de middagpauze op te wekken.*

*Verder is het geleverde cursus materiaal van zeer goede kwaliteit. Besproken onderwerpen komen ruim aan de orde, en worden duidelijk weergegeven waardoor het perfect mogelijk is de behandelde materie in je eigen omgeving en op je eigen tempo nog 's na te bouwen.*

*Kortom, een dikke proficiat voor je presentatie, inhoud en verzorging van de trainingen."*

Karel Cox, Budel Zink

*"Waar Delphi en het internet elkaar kruisen, daar vind je Dr.Bob om de juiste richting aan te duiden"*

Stefan Cruysberghs & Guy Sillen (België)

*"De Delphi Clinics voorkomen weken of zelfs maanden uitzoekwerk, dus is de investering snel terugverdiend"*

Erik van Bilsen

*"Deze Delphi clinics, gegeven door een doorwinterde Delphi specialist, zijn zeer informatief en zullen zeker een meerwaarde bieden bij het ontwikkelen van nieuwe software."*

Jora Soft BVBA (België)

*"Bob Swart heeft in zijn clinics aandacht voor de problematiek van de dagelijkse praktijk. Met zijn expertise, inzicht en behulpzaamheid geeft hij mij een perfect klankbord. Ook na de clinic."*

Hans van de Ven

# Dr. Bob's Delphi 6 Clinics

(met Delphi 7 indien beschikbaar)



12 september 2002

## Delphi COM Development

**Keywords:** Interfaces, COM, Automation, ActiveX, ActiveForms, ADO, MTS/COM+, Active Server Objects

Tijdens deze Delphi Clinic zullen we kennismaken met de wereld van COM en ActiveX. Via interfaces, IUnknown, IDispatch en TInterfacedObject naar COM Objecten (COM Servers en Clients), Automation (Servers en drie soorten clients: Variants, Dispatch en Interfaces), ActiveX/ActiveForms, en ADO (ActiveX Data Objects). Vervolgens gaan we stateless objecten voor Microsoft Transaction Server (MTS) en COM+ ontwikkelen en deployen, en web toepassingen bouwen met Active Server Pages en Objects.

26 september 2002

## dbExpress & DataSnap (MIDAS)

**Keywords:** dbExpress, DataSnap, ClientDataSet, DataSetProvider, Connection components, Deployment

In deze Delphi Clinic behandelen we de cross-platform DataCLX en dbExpress data access laag en de multi-tier architectuur DataSnap (nieuwe naam van MIDAS). De ClientDataSet, DataSetProvider en xxxConnection componenten zullen hierbij uitgebreid aan de orde komen (met aandacht voor TCP/IP Sockets, DCOM, HTTP, SOAP, CORBA en MTS/COM+ connections).

10 oktober 2002

## WebBroker & InternetExpress

**Keywords:** WebBroker, InternetExpress, Debuggen, Deployment

In deze Delphi Clinic starten we met WebBroker als architectuur om CGI of ISAPI/NSAPI web server toepassingen te ontwikkelen, en breiden deze ervaring uit met InternetExpress. Alle WebBroker en InternetExpress componenten (en enkele custom componenten) zullen behandeld worden. Ook het debuggen van web server toepassingen (met de Web App Debugger) komt uitgebreid aan de orde.

24 oktober 2002

## WebSnap Development

**Keywords:** WebSnap, Adapters, WebSnap Custom Adapters, Login/Access, Sessions, Deployment

In deze Delphi Clinic bekijken we WebSnap van alle kanten. De WebSnap Architectuur onderkent een centrale rol voor Adapters. We zullen zien hoe de Adapters werken en alle componenten van de WebSnap tab van het component palette aan de orde laten komen. Daarnaast besteden we aandacht aan login/logout, user access en sessiemanagement, en ontwikkelen een techniek die ook werkt voor CGI toepassingen (en Kylix). Tot slot bekijken we hoe WebSnap achter de schermen in elkaar zit en bouwen WebSnap custom componenten (zoals een TCreditCardAdapter).

7 november 2002

## BizSnap: XML, SOAP & Web Services

**Keywords:** BizSnap, XML Documents, XML Data Binding, XML Mapper, WSDL, SOAP, Web Services, RIO, Clients, Interoperability

Deze Delphi Clinic begint met een overzicht van de XML ondersteuning in Delphi 6, met XML Document Programming, de XML Data Binding Wizard en tenslotte de XML Mapper (een tool waarmee XML documenten naar data packets te transformeren zijn - en terug). Daarna verschuift de aandacht naar SOAP en Web Services. Na een korte introductie in zowel SOAP en WSDL als de toepassing van Web Services, zullen we zien hoe we in Delphi 6.02 bestaande Web Services kunnen gebruiken en nieuwe cross-platform Web Services kunnen maken en deployen.

21 november 2002

## Web Solutions: BizSnap & WebSnap

**Keywords:** e-commerce solutions met WebSnap en BizSnap

Tijdens deze Delphi Clinic bouwen we een "live" multi-tier e-business toepassing met behulp van technieken uit DataSnap, WebSnap en BizSnap (XML & SOAP). We zullen met WebSnap componenten een e-commerce toepassing bouwen met alle toeters en bellen zoals login, registratie nieuwe gebruikers, database access, zoeken, bestellen, winkelwagentje, e-mail support, etc.

5 december 2002

## Delphi .NET Framework Development

**Keywords:** Delphi .NET Framework Development

Tijdens deze Delphi Clinic zullen we zien hoe we Delphi kunnen gebruiken voor .NET Framework Development. Hopelijk al met Delphi 7 en de .NET preview edition...

De lokatie van de Delphi Clinics is de cursusrimte bij Centric PSS in Eindhoven, waar de trainingen worden gegeven op donderdag van 9.00 tot 17.00 uur.

De prijs per clinic is €420 per persoon (excl 19% btw). Bij twee deelnemers van hetzelfde bedrijf geldt een korting van 5% - bij drie of meer deelnemers zelfs 10%.

## LET OP: Speciale Aanbiedingen!

- \* Bij inschrijving vóór 1 augustus krijgt u een t-shirt van Delphi 6 kado - zolang de voorraad strekt!
- \* Bij inschrijving voor drie of meer trainingdagen krijgt u een door mij megeschreven en gesigneerd boek kado (keuze uit Delphi 6 Developer's Guide en Kylix Developer's Guide).
- \* Bij inschrijving voor zes trainingdagen krijgt u de zevende dag kado (dus 7 dagen voor de prijs van zes).

Extra korting is mogelijk bij meer dan vier personen van hetzelfde bedrijf, evenals maatwerk training op lokatie. Zie <http://www.eBob42.com> voor meer informatie, of stuur e-mail naar Bob Swart te [bob@eBob42.com](mailto:bob@eBob42.com)