

# Delphi OplossingsCourant

Vol. 5. No. 2.

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



Helmond, 19-20 mei 2003,

Welkom bij het speciale Conference to the Max nummer van de Delphi OplossingsCourant. Dit nummer wordt uitgedeeld bij de CttM, en staat in het teken van de nieuwe dingen die met name .NET ons brengen zal, waarbij ik eerst dieper inga op de vraag ".NET nu of (nog) niet?", waarin ik tevens aangeef waar (meer) ondersteuning op .NET gebied te vinden is, met name vanuit een Borland invalshoek. Daarna beschrijf ik enkele nieuwe uitbreidingen in de Delphi programmeertaal die we nu reeds in de Delphi for .NET preview command-line compiler aantreffen, maar straks ook in Delphi 8 zullen terugvinden. Voor de niet .NET-fan is er ook een portie "normaal" Delphi met enkele dbExpress tips en een boekbespreking van Mastering Delphi 7 (die deze keer wat op zich heeft laten wachten).

Wie om wat voor reden dan ook niet aanwezig was bij de Conference to the Max kan op de laatste pagina van dit nummer de speciale aanbieding terugvinden voor mijn laatste twee Delphi Clinics van dit seizoen (resp. over Delphi for .NET en IntraWeb). Ook voor niet-CttM bezoekers geldt deze aanbieding: **de tweede Clinic voor de halve prijs** (dus wie eigenlijk alleen in IntraWeb geïnteresseerd is kan voor de halve prijs kennismaken met de Delphi for .NET preview command-line compiler - en vice versa). De enige voorwaarde is dat ik uiterlijk vrijdag 23 mei om 12 uur 's middags de inschrijving ontvangen heb. Dit in verband met het drukken van de juiste hoeveelheid cursusboekjes voor de Delphi for .NET clinic van woensdag 28 mei (vanwege Hemelvaartsdag niet op 29 mei).

Naast Delphi 7 Clinics zijn er ook nog de Delphi Workshops. Deze worden - op aanvraag - voor groepjes vanaf 2 deelnemers (in Helmond) of maximaal 12 (op locatie) georganiseerd, waarbij iedereen zijn eigen computer en versie van Delphi kan gebruiken. De eerste workshops heb ik al achter de rug, en de deelnemers waren zeer enthousiast. Zie <http://www.eBob42.com/workshop> voor details.

## Inhoudsopgave

<b>Welkom</b>	<b>1</b>
<b>Clinics &amp; Evenementen 2003</b>	<b>1</b>
<b>.NET nu of (nog) niet?</b>	<b>2</b>
<b>Delphi taaluitbreidingen</b>	<b>3</b>
<b>dbExpress tips &amp; trucs</b>	<b>5</b>
<b>Boek Review: Mastering Delphi 7</b>	<b>6</b>
<b>Delphi 7 Clinics 2003</b>	<b>6</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)

Wie na lezen van de Delphi OplossingsCourant nog opmerkingen heeft, kan gerust een mailtje sturen naar [bob@eBob42.com](mailto:bob@eBob42.com) - ik stel alle feedback op prijs (dat geldt ook voor vragen of andere ondersteuning die ik kan bieden).

## Delphi Clinics in 2003

In de 1e helft van 2003 zal er elke maand weer een Delphi 7 Clinic plaatsvinden in Eindhoven. De data en onderwerpen van de laatste twee zijn als volgt:

28 mei - **Delphi en .NET Framework / SDK**

26 juni - **Delphi en IntraWeb 5.1 Development**

Zie de achterpagina van deze DOC voor de speciale aanbieding, <http://www.eBob42.com/training> voor meer informatie over de Delphi 7 Clinics, of <http://www.eBob42.com/workshop> voor de Delphi workshops en Custom Clinics.



## Evenementen 2003

Behalve bij de CttM, zal ik ook nog als spreker bij de volgende evenementen aanwezig zijn:

8-10 juni - **DCon 2003**, Londen (UK)

21-26 sept - **EKon 2003**, Frankfurt (DE)

7 okt - **UK-BUG MasterClass**, Londen (UK)

1-5 nov - **BorCon 2003**, San Jose (USA)

# .NET nu of (nog) niet?

Als er iets wel een hype is geworden de afgelopen tijd dan is het wel .NET. Alhoewel Microsoft juist weer een beetje van .NET af begint te stappen (de nieuwe server editie van Windows heet nu gewoon Windows 2003 Server in plaats van Windows .NET Server zoals ze eerst van plan waren). Toch heeft juist de lancering van deze Windows 2003 Server voor een nieuwe impuls gezorgd, al was het dan alleen maar omdat die gepaard ging met de .NET Framework en SDK versie 1.1, en de nieuwe Visual Studio .NET 2003 die nu ook beschikbaar is.

Daarnaast is Borland al ruim een jaar bezig met de Delphi for .NET command-line compiler, maar zal eerst een C# .NET ontwikkelomgeving uit brengen onder de naam C#Builder. Deze zal zo rond 6 juni beschikbaar zijn (zie ook het eerste nummer van Borland .NET Developers' Journal waarin ik al vast wat dieper inga op C#Builder).

Het .NET Framework draait nu nog als een laag bovenop de Win32 API, en biedt operating system en services aan via een soort OO class hierarchy. In plaats van de platte APIs gebruik je in .NET - bij voorkeur - objecten en roep je methoden aan van deze objecten. De .NET classes zijn ingedeeld in namespaces en verzameld in zgn. assemblies (grofweg gezegd de volgende generatie DLL). Omdat nog niet alles uit de Win32 in een .NET assembly terug te vinden is, wordt er nog vaak toch gebruik gemaakt van unsafe, unmanaged code die via DllImport een Win32 API aanroept. Wat dat betreft voelen de huidige versies van het .NET Framework (en SDK) nog een beetje aan als Win32s, dat ook alleen maar de overgang naar puur Win32 aangaf. Het zou me niet verbazen als we over een aantal jaren een puur .NET besturingssysteem hebben, zonder Win32 API er meer onder. Voordeel is dan dat alle toepassingen in safe, managed mode zullen draaien, en er meteen wat minder kans is op beveiligingsproblemen die veroorzaakt worden door buffer overruns, etc. Dat is in mijn ogen trouwens ook een van de redenen om nu al naar .NET over te stappen - op de web server dan - en diep in ASP.NET te duiken.

Zoals ik in de titel van dit stukje al een beetje probeer aan te geven, is in mijn ogen niet de vraag ".NET of niet?" aan de orde, maar eerder de vraag ".NET nu, of nog niet?". Wat dat .NET er zal komen (het is er al), dat staat voor mij wel vast. De enige vraag voor mij is hoe snel het .NET Framework en SDK zal doordringen op de Windows clients en servers en in welke mate de software ontwikkelaars (en daarmee de toekomstige generatie software) het

.NET Framework en SDK zullen omarmen. Wat betreft de Windows servers verwacht ik een snelle groei, de desktop blijft waarschijnlijk nog even achter. Wat dat betreft zie ik het komend jaar meer in investeren rondom ASP.NET en ADO.NET dan .NET GUIs en WinForms. Maar dat is mijn persoonlijke inschatting.

Bestaande ontwikkelingen zullen vaak niet zomaar naar .NET omgezet kunnen worden, maar gelukkig zijn er verschillende manieren om interoperability tussen .NET en Win32 te realiseren. Dat betekent ook dat nieuwe ontwikkelingen nu al in .NET plaats kunnen vinden, om dan toch ook nog in bestaande toepassingen gebruikt te kunnen worden. Wat dat betreft zijn we in mijn ogen duidelijk nog in een overgangsfase tussen unmanaged/unsafe en een volledige .NET wereld.

Wie het avontuur in de .NET wereld niet alleen wil aangaan zal in de SDGN een goede partner vinden - onafhankelijk bovendien, dus zowel aandacht voor Delphi (for .NET) als C# - van Microsoft en Borland. Voor wie nog geen lid is, zie <http://www.sdgn.nl>

Voor een meer internationale benadering kan ik de DotNet Developers Group (DDG) uit Engeland van harte aanbevelen. Met als "trekker" Brian Long, plus de medewerking van Microsoft, en de ervaring en organisatie van de UK Borland User Group (BUG) achter de schermen, liepen de eerste bijeenkomsten gesmeerd. Zie <http://www.richplum.net> voor meer informatie over bijeenkomsten en het lidmaatschap.

Tot slot is er een nieuw blad genaamd Borland .NET Developers Journal dat met name over de Borland ontwikkeltools voor .NET gaat (maar niet door Borland zelf wordt uitgegeven overigens), waarvan het eerste nummer beschikbaar zal zijn vlak voor C#Builder beschikbaar is. Behalve C#Builder zal ook Delphi for .NET daar aan de orde komen.

Uitgegeven door Bridges Publishing, zullen we daar regelmatig artikelen kunnen lezen van Lino Tadros en mijzelf (als contributing editors). Zie de website <http://www.bridgespublishing.com/netjournal> om het eerste nummer (July 2003) gratis te downloaden in PDF formaat.

En ook zelf bied ik de nodig informatie over .NET aan op mijn websites <http://www.drBob42.net> en de Nederlandstalige editie <http://www.drBob42.nl>. Daarnaast organiseer ik al vanaf eind 2002 Delphi for .NET (preview) Clinics, en zal ik binnenkort de eerste reguliere en custom C#Builder Clinics starten. Wie daar - geheel vrijblijvend - meer informatie over wenst te ontvangen kan een mailtje sturen naar [Bob@eBob42.com](mailto:Bob@eBob42.com).

# Delphi Taaluitbreidingen

De Delphi for .NET preview command-line compiler is al een tijd beschikbaar als onderdeel van Delphi 7 Studio. Wie er nog niet mee gespeeld heeft, mist daarmee wellicht een blik in de toekomst wat betreft taaluitbreidingen die we in de volgende versies van Delphi terug kunnen zien. In dit artikel ga ik wat dieper in op enkele van deze uitbreidingen zoals de *visibility specifiers* `strict private` en `strict protected`, `final methods` en tot slot `sealed classes`.

Voor ik begin moet ik er wel bij zeggen dat het net als de Delphi for .NET preview command-line compiler natuurlijk ook een *preview* van de taal (uitbreidingen) betreft - dus alles kan nog wijzigen tussen nu en het moment waarop de volgende Delphi (for .NET) daadwerkelijk beschikbaar is.

## Strict Private en Protected

Vergeleken met C++ of C# is Delphi altijd een beetje te "open" geweest met betrekking tot het gedrag van de "private" en "protected" keywords. In plaats dat private members alleen zichtbaar waren in de class zelf (en protected members in de class en afgeleide classes), kan bij Delphi iedereen uit dezelfde unit toch de private en protected members zien en gebruiken. Te vergelijken met friend classes in C++, alleen is dan altijd de hele unit een vriend van je. Kijk bijvoorbeeld eens naar:

```
type
  MyClass = class
    private
      ZichtbaarInZelfdeUnit: Integer;
    end;
```

Zoals ik al in de naam aangeef, is het veld wel als private gedefinieerd, maar toch zichtbaar voor alles en iedereen in dezelfde unit. Wie toch velden echt wil verbergen zal ofwel per unit maar één class kunnen gebruiken (en verder helemaal niks), of kan in de Delphi for .NET preview command-line compiler gebruik maken van "strict private".

```
type
  MyClass = class
    strict private
      AlleenZichtbaarInZelfdeClass: Integer;
    end;
```

Let op: het gedrag van de "oude" private verandert niet, zodat bestaande code gelukkig niet gebroken zal worden.

Op dezelfde manier is alles wat als protected wordt aangegeven toch zichtbaar in dezelfde unit, maar als je strict protected gebruikt kan alleen de zelfde class of afgeleide classes bij het veld of de methode.

## Final Methods

Behalve de strictere definitie van private en protected bevat de CLR (Common Language Runtime) van .NET ook een eigenschap om aan te geven dat een methode van een class niet langer overruled mag worden. Het is als het ware de laatste verbijzondering, en de methode is "af". Dergelijke methodes worden "final" genoemd, en de Delphi for .NET preview command-line compiler ondersteunt nu ook het final keyword, zoals hieronder gedemonstreerd:

```
type
  BaseClass = class
    procedure Operate; virtual;
  end;

type
  DerivedClass = class(BaseClass)
    procedure Operate; override; final;
  end;
```

De methode Operate van DerivedClass is met het keyword final gedefinieerd, en kan niet in verdere afgeleide classes nog overruled worden. Wie dat wel probeert zal de foutmelding "Error: Cannot override a final method" te zien krijgen.

Wie toch nog een procedure Operate in een afgeleide class wil maken, zal dit moeten doen door override weg te laten (eventueel virtual is wel toegestaan), maar moet beseffen dat hiermee de oorsponkelijke methode Operate in feite wordt verborgen:

```
type
  AnotherClass = class(DerivedClass)
    procedure Operate; virtual;
  end;
```

Dit geeft de duidelijke warning "Method 'Operate' hides virtual method of base type 'DerivedClass'". Wie expliciet wil aangeven dat er over nagedacht is (en de warning dus niet nodig is), kan ook het keyword reintroduce gebruiken. Effect is hetzelfde, alleen volgt er geen warning meer.

Het gebruik van de oorsponkelijke methode Operate kunnen we ook nog voor elkaar krijgen, door via inherited te zoeken naar een methode van dezelfde naam (en signature) in de parent classes. Oftewel de implementatie van de nieuwe Operate kan als volgt:

```
procedure AnotherClass.Operate;
begin
  inherited;
  // calls original Operate from DerivedClass
  ...
end;
```

Wie denkt OO hiermee voor de gek te (kunnen) houden komt bedrogen uit, want de polymorphy keten is wel degelijk doorbroken. Als je een variabele van type `DerivedClass` initialiseert als instantie van `AnotherClass`, dan zal toch alleen de `Operate` van `DerivedClass` gevonden worden. Het `final` keyword maakt dus een eind aan de polymorphy keten, en zo hoort het ook.

## Sealed Classes

Een laatste manier om het polymorphisch gedrag van classes kort te sluiten is het concept van een sealed class: een afgesloten class waar geen andere classes meer van afgeleid mogen (en kunnen) worden:

```
type
  FinalClass = class sealed(DerivedClass)
    procedure Start; override;
  end;
```

Wie toch probeert van `FinalClass` iets af te leiden krijgt de foutmelding "Error: Cannot extend sealed class 'FinalClass'", zoals verwacht.

Als je in de RTL source code van de Delphi for .NET preview command-line compiler kijkt dan kun je op één plek al een sealed class vinden, namelijk in de unit `Borland.Delphi.Variants.pas` met de definitie `TVariant = class sealed;`

Ik heb nog geen final methods aangetroffen.

## Conclusie

In dit artikel heb ik enkele (voorlopige) taaluitbreidingen van de Delphi for .NET preview command-line compiler laten zien die we zeer waarschijnlijk in de toekomstige versies van Delphi zullen terugzien..

Als laatste nog een korte demo met alle features:

```
program eBob42;
{$APPTYPE CONSOLE}

type
  BaseClass = class
    strict private
      FCounter: Integer;
    public
      property Counter: Integer
        read FCounter write FCounter;
    public
      procedure Start; virtual;
      procedure Operate; virtual;
  end;

type
  DerivedClass = class(BaseClass)
    strict protected
      FLevel: Integer;
    public
      procedure Start; override; final;
      procedure Operate; override; final;
  end;
```

```
type
  FinalClass = class sealed(DerivedClass)
    // procedure Start; override;
    // Error: Cannot override a final method
    procedure Start; virtual;
    // Warning: Method 'Start' hides virtual
    // method of base type 'DerivedClass'
    procedure Operate; reintroduce;
  end;

//ErrorClass = class(FinalClass)
//Error: Cannot extend sealed class 'FinalClass'

procedure BaseClass.Start;
begin
  FCounter := 1;
end;

procedure BaseClass.Operate;
begin
  writeln('BaseClass.Operate');
end;

procedure DerivedClass.Start;
begin
  inherited;
  Counter := Counter + 1;
  FLevel := 2
end;

procedure DerivedClass.Operate;
begin
  inherited;
  writeln('DerivedClass.Operate');
end;

procedure FinalClass.Start;
begin
  inherited;
  Counter := Counter + 1;
  FLevel := 3
end;

procedure FinalClass.Operate;
begin
  inherited Operate; // Operate from DerivedClass
  writeln('FinalClass.Operate');
end;

var
  F: FinalClass;
  G: DerivedClass;
begin
  F := FinalClass.Create;
  try
    F.Start;
    F.Operate;
    writeln(F.Counter); // 3
  finally
    F.Free;
  end;

  G := FinalClass.Create;
  try
    G.Start;
    G.Operate;
    writeln(G.Counter); // 2
  finally
    G.Free;
  end;
end.
```

## dbExpress tips & trucs

Zoals iedereen inmiddels wel heeft gehoord, is de Borland Database Engine (BDE) niet meer de aangewezen manier om vanuit Delphi met database tabellen te praten. De BDE is bevroren, en SQL Links (naar de echte DBMS systemen zoals Oracle, DB2 en SQL Server) is zelfs deprecated. Wat wil zeggen dat er geen ondersteuning meer op gegeven wordt. De aangewezen opvolger van de BDE (en SQL Links) is dbExpress. Destijds geïntroduceerd bij Delphi 6, en inmiddels beschikbaar in Delphi, C++Builder en Kylix, vormt het een cross-platform data access laag naar InterBase, DB2, Oracle, SQL Server, MySQL en nog veel meer. Borland maakt drivers voor een aantal databases, maar ook verschillende third-party leveranciers kunnen dbExpress drivers maken.

Helaas zijn er - ook in Delphi 7 - nog steeds een aantal kleine en grote(re) problemen in dbExpress die de overgang en het vertrouwen niet echt ten goede komen. En dat is jammer, want op zich is de manier van werken best fijn, en als je eenmaal weet wat er mis is (en hoe je er omheen kan werken), dan is dbExpress een fijne manier om je databases en tabellen te benaderen en te bewerken.

Enkele zaken waar ik in de afgelopen maanden weer tegenaanliep wil ik hier even onder de aandacht brengen. Allereerst het gebruik van TSQLMonitor, het component dat een logfile kan opleveren van het verkeer tussen de Delphi toepassing en de database zelf - waar de dbExpress driver precies tussen zit. TSQLMonitor heeft een Filename property, alsmede een Active en AutoSave property die je kan gebruiken om te sturen wanneer er naar de logfile geschreven moet worden. Wie liever zelf zijn logfile indeelt, kan gebruik maken van de OnTrace en OnLogTrace event handlers. De eerste wordt aangeroepen - voor elke message van de dbExpress driver naar de database - vlak voordat de message aan de TraceList wordt toegevoegd. OnLogTrace wordt aangeroepen meteen nadat de message aan de TraceList is toegevoegd, maar nog (vlak) voordat deze naar de logfile wordt weggeschreven. In beide gevallen is het mogelijk om de inhoud van de message te bewerken. Hiervoor krijgen we een speciale CBIInfo parameter mee, waarin we ook een veld eTraceCat terugvinden waarin de category van de message zou moeten worden aangegeven. Dat is handig om onderscheid te maken tussen prepare, execute, error, connect, transactie, vendor, data in/uit of andere soorten berichten. Helaas, helaas, de waarde van eTraceCat is altijd 255 en daardoor minder bruikbaar als "filter" voor de logfile.

Wie overigens de TSQLMonitor niet kan vinden, gebruikt waarschijnlijk Delphi 7 Professional, waar hij officieel niet in zit. Toch kun je hem wel tevoorschijn toveren, omdat het component namelijk in de unit SqlExpr.pas gedefinieerd is, waar ook de rest van dbExpress in zit. Bij Delphi 7 Professional heeft men TSQLMonitor dus niet geregistreerd, maar is hij er wel. Dit kun je zichtbaar maken door het component als volgt te registreren:

```
unit TSQLMonitorReg;  
interface  
uses  
    SqlExpr;  
  
procedure Register;  
  
implementation  
uses  
    Classes;  
  
procedure Register;  
begin  
    RegisterComponents('dbExpress', [TSQLMonitor])  
end;  
  
end.
```

Of Borland hier blij van wordt weet ik niet, maar laten ze eerst de eTraceCat maar eens fixen en dan zien we wel weer verder.

### Berekende truc

Als laatste "grap" kwam ik er laatst weer eens achter dat de OnCalculateField event handler in een dbExpress dataset (dus TSQLQuery, TSQLTable, TSQLStoredProc of TSQLDataSet) niet voor het eerste record uit de dataset wordt aangeroepen. Wel voor de rest, maar voor het eerste record blijven de calculated fields dus gewoon helemaal leeg. Het advies hier is om de code voor het calculated field niet op dbExpress niveau maar pas op het ClientDataSet niveau uit te voeren.

Dat is echter niet altijd mogelijk, en wie - geheel voor eigen risico - dit probleem anders uit de wereld wil helpen, kan in unit SqlExpr.pas (uit de Delphi7\Source\Vcl directory) de regel 3353

```
if EOF = True then
```

veranderen in

```
if (EOF = True) and  
    ((RecNo >= RecordCount) or  
    (RecordCount < 1)) then
```

Het gebruik van RecordCount is riskant, omdat er nu een extra query wordt uitgevoerd (een select count), maar je weet nu in ieder geval wel zeker dat het eerste record aan de beurt komt.

Bij mij werkt het, maar gebruik aub alleen op eigen risico (totdat Borland met een echte fix komt).

## Review: Mastering Delphi 7

Het lijkt er op dat nieuwe versies van Delphi steeds minder vaak ook nieuwe boeken tot gevolg hebben. Waren er nog enkele Delphi 6 specifieke boeken, ik ken op dit moment nog maar één specifieke Delphi 7 titel. Dat kan natuurlijk ook alles te maken hebben met het feit dat er in Delphi 7 toch vrij weinig echt nieuwe dingen zijn toegevoegd - het merendeel was al beschikbaar als third-party product, maar zit nu in de Delphi 7 Studio doos.

Marco Cantù is dan ook de enige schrijver die vanaf de eerste versie er elke keer weer in geslaagd is om een Mastering Delphi boek uit te brengen, en aan de top van de Delphi bestsellerslijst te zetten. Ook deze keer is het boek Mastering Delphi 7 weer van een indrukwekkende omvang: meer dan 1000 pagina's, maar geen CD met goodies of de elektronische versie van het boek (wat zoeken vaak kan helpen), zoals wel bij SAMS bijvoorbeeld gedaan wordt.

Mastering Delphi 7 bestaat uit 25 hoofdstukken en 3 korte appendices. Om met de laatste te beginnen, in de appendices krijg je een overzicht van de gratis Delphi tools en utilities die door Marco of andere personen zijn geschreven en handig in het gebruik zijn. De laatste appendix bevat verwijzingen naar Marco's gratis e-boeken Essential Pascal, Essential Delphi en Delphi Power Book. Met name Essential Delphi is de plek waar oude hoofdstukken uit voorgaande edities van Mastering Delphi een plek vinden. Dat is de reden waarom Mastering Delphi 7 weer met nieuwe onderwerpen is uitgerust zonder dat het boek inmiddels 2000 pagina's dik is. De basis principes en beginnende hoofdstukken zijn in Essential Delphi terug te vinden (wat daarmee een goede manier is om iemand met Delphi of Pascal te laten kennismaken, waarna Mastering Delphi 7 het aangewezen vervolg is).

Van de 25 hoofdstukken zijn er een aantal geheel nieuw, namelijk hoofdstuk 11 over *Modeling and OOP Programming* (met ModelMaker), hoofdstuk 18 over *Reporting with Rave*, hoofdstuk 21 over *Web Programming with IntraWeb*, en de laatste twee hoofdstukken (24 en 25) over *The Microsoft .NET Architecture from the Delphi Perspective*, en *Delphi for .NET Preview: The Language and the RTL*. Bij elkaar zo'n 110 nieuwe pagina's in 5 nieuwe hoofdstukken. En de overige 20 hoofdstukken zijn aangepast aan Delphi 7, wat soms ook tot flink wat nieuwe stukken tekst heeft geleid.

---

**Auteur:** Marco Cantù (plus enkele gastschrijvers)

**Uitgever:** Sybex

**ISBN:** 0-7821-4201-X

**Prijs:** \$ 59.99, 1010 pagina's (geen CD-ROM)

## Dr. Bob's Delphi Clinics

De eerstvolgende twee Delphi Clinics worden op resp. woensdag 28 mei en donderdag 26 juni gegeven.

28 mei 2003

### Delphi .NET Framework Development

---

**Keywords:** Delphi .NET Framework Development, ASP.NET, VCL for .NET

Tijdens deze Delphi Clinic zullen we zien hoe we Delphi kunnen gebruiken voor .NET Framework Development met de Delphi for .NET preview command-line compiler. Onderwerpen zijn het gebruik van COM in .NET (en vice versa), .NET assemblies, van VCL naar VCL for .NET, native WinForms, databases benaderen met ADO.NET, en tot slot Delphi als scripting taal bij ASP.NET voor web forms en web services.

26 juni 2003

### Web Development met IntraWeb 5.1

---

**Keywords:** IntraWeb, Page Mode, Application Mode, Login, Sessions, State Management, IntraWeb Components, Deployment

In deze Delphi Clinic bekijken we IntraWeb van alle kanten. Nadat WebBroker door InternetExpress (Delphi 5) en daarna met WebSnap (Delphi 6) is uitgebreid, bleven vele ontwikkelaars toch zitten met een gevoel van "kan het niet wat meer op de Delphi manier?" Met IntraWeb - een third-party tool van AtoZedSoftware - die bij Delphi 7 zit, kunnen we dit nu eindelijk: web server toepassing bouwen zoals je normale Delphi programma's bouwt. Tijdens deze Delphi Clinic laat ik zien hoe we met IntraWeb nieuwe web server toepassingen kunnen maken, hoe we die kunnen deployen op een web server (welke vormen wel, welke vormen niet), en ook hoe we met IntraWeb onze bestaande WebBroker of WebSnap toepassingen kunnen uitbreiden met IntraWeb "Page Mode" functionaliteiten (zodat we niet nog een keer bestaande investeringen hoeven weg te gooien of opnieuw hoeven te beginnen).

De meeste aandacht gaat echter uit naar IntraWeb in "Application Mode", de IntraWeb componenten (en hoe we die zelf kunnen bouwen) en State Management binnen IntraWeb wat geheel transparant is (maar ik laat zien hoe en wat er achter de schermen gebeurt).

We gebruiken de nieuwste versie van IntraWeb 5.1.

De locatie van de Delphi Clinics is de cursusruimte 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 Aanbieding!

\* Bij inschrijving voor beide trainingdagen krijgt u de tweede dag voor de halve prijs (dus €210 euro als u alleen komt, en nog minder bij meer personen van hetzelfde bedrijf).

**Inschrijven voor deze speciale aanbieding kan tot vrijdag 23 mei, 12.00 uur 's middags!**