

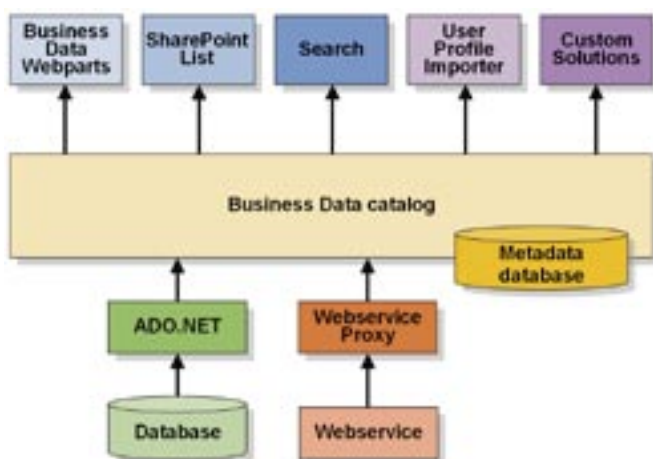
De business data catalog

DATA UIT BEDRIJFSAPPLICATIES ONTSLUITEN

Met de komst van Microsoft Office SharePoint Server 2007 en de Business Data Catalog (BDC) biedt Microsoft een oplossing om data uit bedrijfsapplicaties op een effectieve manier te ontsluiten. In dit artikel behandelt de auteur de architectuur van de BDC en laat zij zien hoe data uit bedrijfsapplicaties met behulp van de BDC in Office SharePoint Server kunnen worden gebruikt.

Het huidige bedrijfsklimaat met zijn nadruk op het vergroten van marges, het continu verlagen van kosten en meer doen met minder, maakt het voor alle onderdelen van een organisatie belangrijk om aan uniforme doelstellingen te werken. Hierbij is het essentieel dat de juiste mensen de juiste data tot hun beschikking hebben. Een intranetplatform is tegenwoordig de ideale plek om deze data zichtbaar te maken. Bedrijfsprocessen worden echter vaak bepaald door een mix van 'harde' en 'softe' data. Met 'harde' data wordt informatie in bedrijfsapplicaties bedoeld; 'softe' data hebben daarentegen betrekking op informatie geproduceerd via desktopapplicaties, zoals Microsoft Office. De Office-bestanden worden vandaag de dag al veelvuldig op een intranet gedeeld, maar de weergave van data uit bedrijfsapplicaties was tot nu toe vaak een tijdrovende en dure aangelegenheid.

Met behulp van Microsoft Office SharePoint Server 2007 (MOSS) en de Business Data Catalog biedt Microsoft hiervoor een oplossing. Om data uit bedrijfsapplicaties op een effectieve manier te ontsluiten, wordt de datastructuur van de applicatie beschreven met behulp van het BDC-metadatumodel. Deze beschrijving resulteert in een applicatiedefinitie. Met behulp van de applicatiedefinitie kunnen de businessdata uit bijvoorbeeld een SAP- of een Siebel-systeem worden gepubliceerd in een Sharepoint-omgeving. In dit artikel zal ik een tipje van de sluier van de BDC oplichten. Ik vertel allereerst iets over de architectuur van de BDC, gevolgd door het metadataschema en de objectmodellen. Daarna laat ik zien hoe data uit bedrijfsapplicaties met behulp van de BDC in MOSS 2007 kunnen worden gebruikt. Ten slotte geef ik nog een klein voorbeeld van het gebruik van de BDC in custom applicaties.



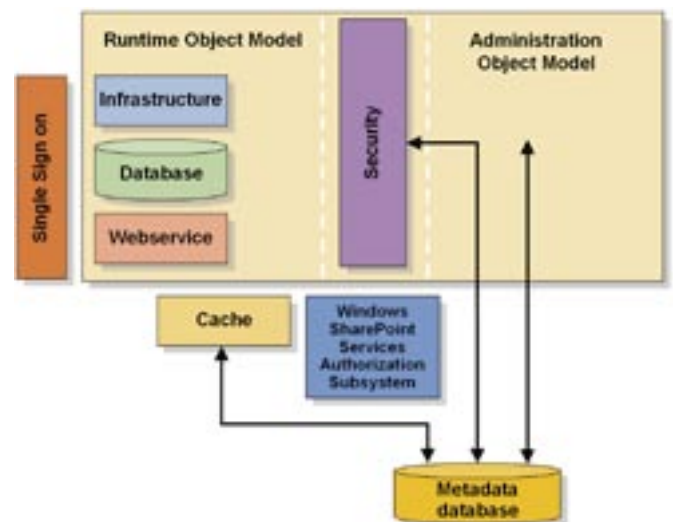
Afbeelding 1. Schematische weergave van de architectuur van de BDC

Architectuur

Shared Service Providers (SSP's) zijn een nieuw onderdeel van Microsoft Office SharePoint Server 2007 en vormen een belangrijk onderdeel van de architectuur. Een SSP bestaat uit een set met services die centraal geconfigureerd kunnen worden. Vervolgens kunnen de services in de SSP door meer MOSS- en WSS-sites in een farm gebruikt worden. De MOSS-features die beschikbaar zijn via SSP's, zijn User Profiles, Audiences, Personal Sites, Excel Services, de Business Data Catalog en Search.

De BDC is dus een MOSS 2007 Shared Service. De metadata van een BDC-applicatie worden opgeslagen in een SSP-database die de metadata-repository genoemd wordt. De BDC kan de data van bedrijfsapplicaties waarvan de data bereikbaar zijn, via een database/ADO.Net of een webservice ontsluiten. Deze zijn vervolgens beschikbaar in Sharepoint en in custom applicaties. Afbeelding 1 toont een schematische weergave van de architectuur van de BDC.

Met behulp van het objectmodel van de BDC kunnen gegevens van in de BDC geïmporteerde bedrijfsapplicaties ook in custom applicaties gebruikt worden. Dit objectmodel bestaat uit twee delen, het Runtime-objectmodel en het Administration-objectmodel. Door bij het bouwen van custom applicaties gebruik te maken van de BDC zijn de data uit diverse bedrijfsapplicaties via een consistente en objectgeoriënteerde programmeerinterface te benaderen. Afbeelding 2 toont de low-level architectuur van de BDC.



Afbeelding 2. De low-level architectuur van de BDC

De metadatadatabase, ofwel metadata repository, wordt gebruikt voor het opslaan van de applicatiedefinities. In de applicatiedefinitie wordt de structuur van de data uit bedrijfsapplicaties beschreven met behulp van XML. De XML is opgebouwd aan de hand van het metadata-schema. Voor elke applicatie die ontsloten moet worden via de BDC, worden de metadata gedefinieerd door de entiteiten en de methoden te beschrijven waar de applicatie gebruik van maakt. Stel je wilt de data uit de AdventureWorks-database ontsluiten in Sharepoint. Je kunt er voor kiezen de entiteiten Customer, SalesOrder en Product te beschrijven. Daarnaast definieer je met behulp van SQL-statements de metadata voor methoden die gebruikt worden om data uit de database te lezen. Codevoorbeeld 1 laat een deel van de XML zien, waarmee de entiteit Product en een methode voor het ophalen van producten gedefinieerd wordt.

De BDC maakt geen kopie van de data uit de bedrijfsapplicaties. De getoonde gegevens komen direct uit de datastore van de bedrijfsapplicatie. Om de data toch snel aan te kunnen bieden, worden de metadata gecached. Elke minuut wordt gekeken of de metadata veranderd zijn. Is dit het geval, dan wordt de cache opnieuw gevuld. In een Sharepoint-farm met meer servers kan het hierdoor tot één minuut duren voordat alle servers de aangepaste data tonen. Met behulp van het objectmodel is het mogelijk caching op bepaalde delen van het metadatamodel uit te zetten. Dit is nuttig voor objecten die zelden gebruikt worden.

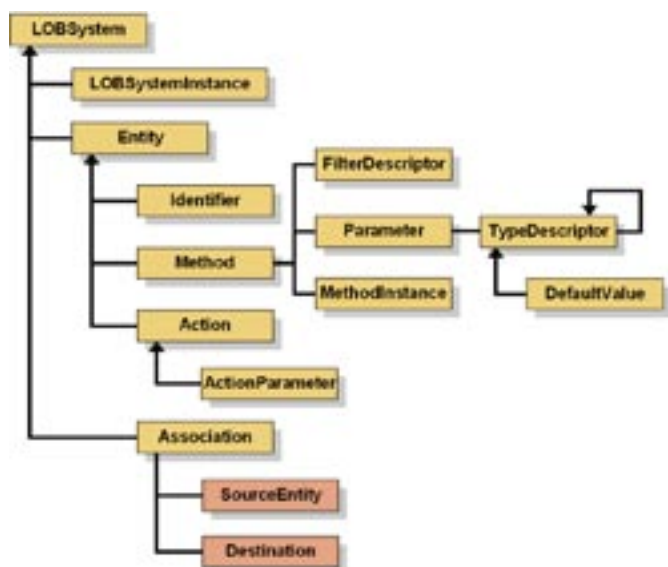
Metadataschema

Het metadatamodel beschrijft de datastructuur van de bedrijfsapplicatie. De belangrijkste objecten hierin zijn:

- **LobSystem:** een data-source (zoals Siebel of een CRM-system)
- **Entity:** een entiteit (bijvoorbeeld klant of product)
- **Method:** een actie die uitgevoerd kan worden op een entiteit (bijvoorbeeld GetProducts)
- **Association:** een relatie tussen entiteiten (bijvoorbeeld klanten hebben orders en orders bestaan uit producten).

Het XML-schema en de metadata-objectmodellen volgen elkaar en het metadatamodel nauwkeurig. Afbeelding 3 toont het metadatamodel en in codevoorbeeld 2 is de definitie van een LobSystemInstance te zien.

In de LobSystemInstance worden de gegevens beschreven die de BDC nodig heeft om verbinding te maken met de database van een bedrijfsapplicatie. In dit voorbeeld wordt de database AdventureWorks beschreven, die met behulp van integrated security wordt benaderd.



Afbeelding 3. Het metadatamodel

Runtime-objectmodel

Het Runtime-objectmodel is ontworpen om gebruikt te worden door BDC-clients en applicaties. Met behulp van het Runtime-objectmodel zijn de data uit verschillende bedrijfsapplicaties via een objectgeoriënteerde interface te benaderen. Het ophalen van data uit een SAP-applicatie gaat op dezelfde manier als het ophalen van data uit een Siebel-applicatie. De BDC voert overigens zelf geen methoden uit en regelt ook geen infrastructuur, maar zorgt er slechts voor dat er gebruik gemaakt wordt van de juiste ADO.NET-provider voor databases en van de juiste webservices-proxy voor webservices. De BDC heeft zelf ook een webservice waarmee het Runtime-objectmodel beschikbaar gemaakt wordt. Hierdoor is het mogelijk de metadata repository van buitenaf te benaderen.

Administration-objectmodel

Het beschikbaar maken van data uit een bedrijfsapplicatie door de BDC kan niet alleen door het beschrijven van de metadata met behulp van XML in de applicatiedefinitie en het importeren van deze applicatiedefinitie. De metadata van de bedrijfsapplicatie kunnen ook beschreven en geïmporteerd worden in de BDC met behulp van het Administration-objectmodel. Dit maakt het mogelijk tools te schrijven, waarmee het eenvoudig wordt om metadata te beschrijven en te onderhouden.

Business Data in MOSS 2007

Data uit bedrijfsapplicaties, die ontsloten zijn met behulp van de BDC, zijn direct beschikbaar in Microsoft Office SharePoint Server 2007. De data kunnen worden gebruikt in Business Data-webparts, lijsten, Business Data-actions, Business Data Search en Business Data in user-profiles.

```

<Entities>
  <!-- Naam van de entiteit. -->
  <Entity EstimatedInstanceCount="10000" Name="Product">
    <Properties>
      <!-- Title is de default weergave waarde van deze entiteit.
      Deze wordt bijvoorbeeld als titel van zoekresultaten gebruikt.-->
      <Property Name="Title" Type="System.String">Name</Property>
    </Properties>
    <Identifiers>
      <!-- Primary key voor Business Data Catalog Entiteiten. -->
      <Identifier TypeName="System.Int32" Name="ProductID" />
    </Identifiers>
    <Methods>
      <!-- Naam van de methode -->
      <Method Name="GetProducts">
        <Properties>
          <!-- SQL query of naam van de stored procedure -->
          <Property Name="RdbCommandText" Type="System.String">
            SELECT ProductID, Name, ProductNumber
            FROM Production.Product WHERE
            (ProductID >= @MinProductID) AND
            (ProductID <= @MaxProductID) AND (Name LIKE @Name)
          </Property>
          <!-- RdbCommandType is Text of StoredProcedure-->
          <Property Name="RdbCommandType"
            Type="System.Data.CommandType,
            System.Data, Version=2.0.0.0,
            Culture=neutral,
            PublicKeyToken=b77a5c561934e089">
            Text
          </Property>
        </Properties>
      </Method>
    </Methods>
  </Entity>
</Entities>
  
```

Codevoorbeeld 1.

ID	Name	Product Number
1	AdventureWorks	AW-0001
2	AdventureWorks	AW-0002
3	AdventureWorks	AW-0003
4	AdventureWorks	AW-0004
5	AdventureWorks	AW-0005
6	AdventureWorks	AW-0006
7	AdventureWorks	AW-0007
8	AdventureWorks	AW-0008
9	AdventureWorks	AW-0009
10	AdventureWorks	AW-0010
11	AdventureWorks	AW-0011
12	AdventureWorks	AW-0012
13	AdventureWorks	AW-0013
14	AdventureWorks	AW-0014
15	AdventureWorks	AW-0015
16	AdventureWorks	AW-0016
17	AdventureWorks	AW-0017
18	AdventureWorks	AW-0018
19	AdventureWorks	AW-0019
20	AdventureWorks	AW-0020

Afbeelding 4. Een screenshot van een webpart Business Data List

MOSS 2007 wordt geleverd met vijf Business Data-webparts:

- **Business Data List webpart** voor het weergeven van een lijst van data uit de bedrijfsapplicatie. In het webpart-panel kan geconfigureerd worden welke entiteit uit welke geregistreerde applicatie getoond moet worden. Afbeelding 4 laat een screenshot zien van een Business Data List-webpart met data uit de AdventureWorks-producttabel. Afbeelding 4 toont een lijst met alle producten, maar het is ook mogelijk de Business Data List-webpart zo te configureren dat een gebruiker in de data kan zoeken. Afbeelding 5 laat de webpart zien, nadat een gebruiker in de 'name'-kolom van de productenlijst op 'bike' heeft gezocht. De Business Data List-webpart kan door middel van een webpart-connectie verbonden worden met de webpart Business Data Item. In de list-webpart kan een item geselecteerd worden, waarna de details in de item-webpart getoond worden.
- **Business Data Item-webpart** voor het weergeven van details. In afbeelding 6 is een Business Data Item-webpart te zien, waarin het geselecteerde item uit de Business Data List-webpart van afbeelding 4 wordt weergegeven. Het is ook mogelijk om de Business Data Item-webpart te configureren om een losstaand item uit de BDC te tonen.
- **Business Data Related List-webpart** voor het weergeven van een lijst gerelateerd aan een selectie in een Business Data List-webpart. Deze webpart kan gebruikt worden voor het tonen van één op meer relaties: bijvoorbeeld het tonen van alle orders van een in de Business Data List-webpart geselecteerde klant.
- **Business Data Actions-webpart** voor het weergeven van de acties die gedefinieerd zijn voor een entiteit in de BDC. De webpart Business Data Actions kan net als de webpart Business Data Item gekoppeld worden aan de webpart Business Data List. Afbeelding 7 toont de webpart Business Data Actions.
- **Business Data Item Builder-webpart** voor het creëren van een Business Data Item op basis van een waarde in de querystring. Dit item is vervolgens beschikbaar voor andere webparts op de pagina.

Business Data Actions

Acties kunnen gedefinieerd worden voor entiteiten in de BDC. Acties kunnen worden gebruikt in scenario's waarin data terug-

ID:	1
Name:	Adjustable Race
ProductNumber:	AR-5381

Afbeelding 6. Een Business Data Item-webpart

Waar Profile
Gebruik en Lijst

Afbeelding 7. De webpart Business Data Actions

ID	Name	Product Number
875	Adjustable Race	AR-5381
877	Adjustable Race	AR-5381
878	Adjustable Race	AR-5381
879	Adjustable Race	AR-5381
880	Adjustable Race	AR-5381

Afbeelding 5. Een webpart nadat een gebruiker op 'bike' heeft gezocht

geschreven worden naar de bedrijfsapplicatie. Hierbij kun je denken aan een user-profile, waarin gegevens worden aangepast, die vervolgens ook automatisch in het SAP-systeem worden aangepast. Het kan ook een lijst zijn met klanten uit een CRM-systeem, waarbij het adres van een klant in het CRM-systeem kan worden aangepast. Acties worden geïmplementeerd als links waaraan je, indien gewenst, parameters met data uit de bedrijfsapplicatie kunt meegeven. Dit betekent dat een actie ook kan bestaan uit het sturen van een e-mail of het openen van een website. Het is ook mogelijk om bijvoorbeeld op LIVE te zoeken met een waarde uit de bedrijfsapplicatie. In codevoorbeeld 3 is de XML uit de applicatiedefinitie te zien, waarmee deze actie gedefinieerd wordt.

Business Data in lijsten

In MOSS 2007 wordt een nieuw soort kolom geïntroduceerd voor het gebruik in lijsten, het type van deze kolom is 'Business Data'. In deze kolom kunnen gegevens getoond worden uit een bedrijfsapplicatie die geregistreerd is in de BDC. Met behulp van dit type kolom kunnen metadata uit een bedrijfsapplicatie worden toegevoegd aan documenten, kan een dropdown-lijst gevuld worden met data uit een database, en kunnen data uit bedrijfsapplicaties gebruikt worden in Office-documenten.

Business Data Search

Met behulp van de BDC kan de search in Microsoft Office SharePoint Server 2007 alle data in de bedrijfsapplicaties indexeren die geregistreerd zijn in de BDC. Hierdoor is een full-text search op deze data mogelijk. Dit betekent dat een eindgebruiker dankzij MOSS 2007 Search Center kan zoeken door data in een bedrijfsapplicatie. De resultaten kunnen getoond worden tussen de Sharepoint-resultaten, maar het is ook mogelijk een apart tabblad in te richten voor het zoeken in de businessdata. In afbeelding 8 is een

```

<LobSystemInstances>
  <!--De naam van de applicatiedefinitie -->
  <LobSystemInstance Name="AdventureWorksArticleSample">
    <Properties>
      <!--PassThrough, RevertToSelf, Credentials, WindowsCredentials -->
      <Property Name="AuthenticationMode"
        Type="System.String">RevertToSelf</Property>
      <!--SqlServer, OLEDB, Oracle, ODBC -->
      <Property Name="DatabaseAccessProvider"
        Type="System.String">SqlServer</Property>
      <!--De naam van de server waar de te registreren database staat -->
      <Property Name="RdbConnection Data Source"
        Type="System.String">JouwAdventureWorksServerNaam</Property>
      <!--De naam van de database -->
      <Property Name="RdbConnection Initial Catalog"
        Type="System.String">AdventureWorks</Property>
      <Property Name="RdbConnection Integrated Security"
        Type="System.String">SSPI</Property>
      <Property Name="RdbConnection Pooling"
        Type="System.String">>false</Property>
    </Properties>
  </LobSystemInstance>
  
```

Codevoorbeeld 2.

```

using System;
using Microsoft.Office.Server.ApplicationRegistry.MetadataModel;
using Microsoft.Office.Server.ApplicationRegistry.Infrastructure;
using Microsoft.Office.Server.ApplicationRegistry.Runtime;

namespace BusinessDataCatalog.Samples
{
    class GetBusinessData
    {
        const string yourSSPName = "SharedServices1";

        static void Main()
        {
            DisplayLOBSystemsInBDC();
            Console.WriteLine("Press any key to exit...");
            Console.Read();
        }

        static void DisplayLOBSystemsInBDC()
        {
            // Setup BDC Instance
            SqlSessionProvider.Instance().SetSharedResourceProviderToUse
                (yourSSPName);

            // Ophalen LobSystemInstances
            NamedLobSystemInstanceDictionary sysInstances =
                ApplicationRegistry.GetLobSystemInstances();
            // Ophalen LobSystemInstance AdventureWorksSample
            LobSystemInstance sysInstance =
                sysInstances["AdventureWorksSample"];
            // Ophalen entities dventureWorksSample LobSystemInstance
            NamedEntityDictionary entities = sysInstance.GetEntities();
            // Ophalen entity met de naam Products
            Entity bdcEntity = entities["Products"];
            // Ophalen Finder methode van de Products entity
            MethodInstance methInstance =
                bdcEntity.GetFinderMethodInstance();
            // Ophalen instances uit Products entity
            IEntityInstanceEnumerator bdcEntityInstanceEnumerator =
                (IEntityInstanceEnumerator)bdcEntity.Execute(methInstance,
                    sysInstance);
            // Loop door alle instances in de entity
            while (bdcEntityInstanceEnumerator.MoveNext())
            {
                IEntityInstance IE =
                    bdcEntityInstanceEnumerator.Current;
                // Loop door alle velden uit de instance
                foreach (Field f in bdcEntity.GetFinderView().Fields)
                {
                    Console.WriteLine(IE[f]);
                }
                Console.WriteLine("");
            }
        }
    }
}

```

Codevoorbeeld 4.

screenshot te zien van de zoekresultaten na het zoeken op 'bike' in de database AdventureWorks.

Business Data in User Profiles

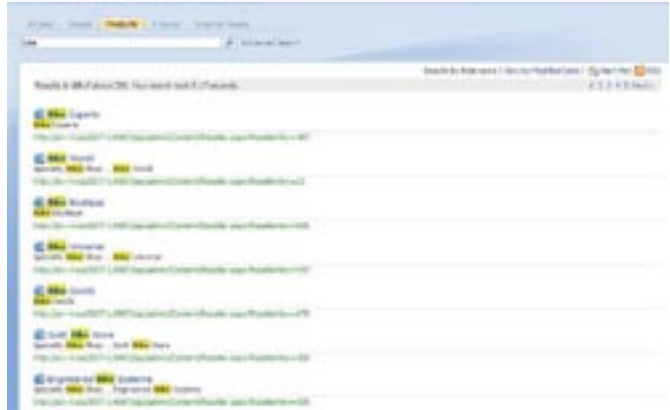
Informatie over personen in een bedrijf wordt vaak niet opgeslagen in Active Directory. Daardoor was het voorheen niet mogelijk alle data van personen in de user-profiles in Sharepoint te tonen. Ook moesten de data op twee plekken worden bijgehouden. Met behulp van de BDC is het nu mogelijk data uit bijvoorbeeld SAP of Siebel in de user-profiles te tonen, zonder dat je hiervoor code hoeft te schrijven.

```

<Actions>
  <Action Name="Search on LIVE"
    Position="1"
    IsOpenedInNewWindow="true"
    Url="http://www.live.com/?q={0}"
    ImageUrl="">
    <ActionParameters>
      <ActionParameter Name="Name" Index="0" />
    </ActionParameters>
  </Action>
</Actions>

```

Codevoorbeeld 3.



Afbeelding 8. Screenshot met de zoekresultaten

Custom applicaties met de Business Data Catalog

De BDC kan niet alleen gebruikt worden voor het ontsluiten van data in Sharepoint. Je kunt ook data uit de BDC gebruiken in custom applicaties. In codevoorbeeld 4 is te zien hoe gegevens uit de BDC kunnen worden gelezen met behulp van het Runtime-objectmodel.

Eenvoudige dataontsluiting

Met behulp van de Business Data Catalog zijn data uit bedrijfsapplicaties op eenvoudige wijze te ontsluiten. Als een applicatie eenmaal in de BDC geregistreerd is, wordt het relatief gemakkelijk om de data in Sharepoint of in een custom applicatie te tonen. Dit zal de informatievoorziening in veel bedrijven verbeteren en bedrijfsprocessen zullen soepeler verlopen. Hiermee wordt tijd en geld bespaard en dat is een uitstekend argument om elke manager ervan te overtuigen dat het een goed idee is om Microsoft Office SharePoint Server 2007 met de Business Data Catalog te implementeren.

Mirjam van Olst werkt als lead developer voor het Sharepoint Competence Center van Macaw (www.macaw.nl). Haar e-mailadres is mirjam.van.olst@macaw.nl

Referenties

Business Data Catalog op MSDN: <http://msdn2.microsoft.com/en-us/library/ms563661.aspx>
 Business Data Catalog Samples op GotDotNet: <http://www.gotdotnet.com/codegallery/codegallery.aspx?id=5e078686-a05c-4a44-a131-88d75e550be8>
 Blog van MVP Luis du Solier over de Business Data Catalog: <http://www.sharepointblogs.com/Idusolier/archive/2006/09/22/12574.aspx>