Deploy GetADobjects assembly

Open SQL Management Studio and connect to the SQL server you will use.

CREATE DATABASE [AD\_DW];

ALTER DATABASE [AD\_DW] SET TRUSTWORTHY ON;

USE [AD\_DW];

GO

CREATE ASSEMBLY [System.DirectoryServices.AccountManagement] from 'C:\Windows\Microsoft.NET\Framework64\v4.0.30319\System.DirectoryServices.AccountManagement.dll' with permission\_set = UNSAFE --Fails if not 64 on 64 bit machines

GO

|  |
| --- |
| *Note – you will get three warnings – but thats expected:*  Warning: The Microsoft .NET Framework assembly 'system.directoryservices.accountmanagement, version=4.0.0.0, culture=neutral, publickeytoken=b77a5c561934e089, processorarchitecture=msil.' you are registering is not fully tested in the SQL Server hosted environment and is not supported. In the future, if you upgrade or service this assembly or the .NET Framework, your CLR integration routine may stop working. Please refer SQL Server Books Online for more details.  Warning: The Microsoft .NET Framework assembly 'system.directoryservices.protocols, version=4.0.0.0, culture=neutral, publickeytoken=b03f5f7f11d50a3a, processorarchitecture=msil.' you are registering is not fully tested in the SQL Server hosted environment and is not supported. In the future, if you upgrade or service this assembly or the .NET Framework, your CLR integration routine may stop working. Please refer SQL Server Books Online for more details.  Warning: The Microsoft .NET Framework assembly 'system.directoryservices, version=4.0.0.0, culture=neutral, publickeytoken=b03f5f7f11d50a3a, processorarchitecture=msil.' you are registering is not fully tested in the SQL Server hosted environment and is not supported. In the future, if you upgrade or service this assembly or the .NET Framework, your CLR integration routine may stop working. Please refer SQL Server Books Online for more details. |

CREATE ASSEMBLY [GetADobjects] FROM 'C:\Temp\GetADobjects.dll' WITH PERMISSION\_SET = UNSAFE;

GO

CREATE PROCEDURE [dbo].[clr\_GetADobjects]

@ADpath [nvarchar](max),

@ADfilter [nvarchar](max),

@MemberList [xml] OUTPUT

WITH EXECUTE AS CALLER

AS EXTERNAL NAME [GetADobjects].[StoredProcedures].[clr\_GetADobjects];

GO

CREATE PROCEDURE [dbo].[clr\_GetADusersPhotos]

@ADpath [nvarchar](max),

@ADfilter [nvarchar](max)

WITH EXECUTE AS CALLER

AS EXTERNAL NAME [GetADobjects].[StoredProcedures].[clr\_GetADusersPhotos]

GO

Test if it is working - get all users:

DECLARE @ADpath nvarchar(64) = 'LDAP://DC=veca,DC=is';

DECLARE @ADfilter nvarchar(64) = '(&(objectCategory=person)(objectClass=user))';

DECLARE @Members XML;

EXEC clr\_GetADobjects @ADpath, @ADfilter, @Members OUTPUT;