﻿using System;

using System.IO;

using System.Runtime.InteropServices;

namespace SqlServerTypes

{

/// <summary>

/// Utility methods related to CLR Types for SQL Server

/// </summary>

public class Utilities

{

[DllImport("kernel32.dll", CharSet = CharSet.Auto, SetLastError = true)]

private static extern IntPtr LoadLibrary(string libname);

/// <summary>

/// Loads the required native assemblies for the current architecture (x86 or x64)

/// </summary>

/// <param name="rootApplicationPath">

/// Root path of the current application. Use Server.MapPath(".") for ASP.NET applications

/// and AppDomain.CurrentDomain.BaseDirectory for desktop applications.

/// </param>

public static void LoadNativeAssemblies(string rootApplicationPath)

{

var nativeBinaryPath = IntPtr.Size > 4

? Path.Combine(rootApplicationPath, @"SqlServerTypes\x64\")

: Path.Combine(rootApplicationPath, @"SqlServerTypes\x86\");

LoadNativeAssembly(nativeBinaryPath, "msvcr120.dll");

LoadNativeAssembly(nativeBinaryPath, "SqlServerSpatial140.dll");

}

private static void LoadNativeAssembly(string nativeBinaryPath, string assemblyName)

{

var path = Path.Combine(nativeBinaryPath, assemblyName);

var ptr = LoadLibrary(path);

if (ptr == IntPtr.Zero)

{

throw new Exception(string.Format(

"Error loading {0} (ErrorCode: {1})",

assemblyName,

Marshal.GetLastWin32Error()));

}

}

}

}