

EIA Sample Project

This sample project illustrates how the EViews Database Extension (EDX) API was used to add support for U.S. Energy Information Administration (EIA) data to EViews 8.1. The EDX interface is an open API that allows anyone to extend EViews to support direct access to any public or private data source. This project assumes some familiarity with the EDX API. The full specification of the EDX interface (including walkthroughs of some simpler examples) is available at

http://www.eviews.com/download/whitepapers/EViews_Database_Extension_Interface.pdf

This project implements two EViews Database Extensions: one that supports online access to EIA data by connecting to the EIA web server and one that supports offline access to EIA data by supporting the EIA bulk download file format.

The project also shows how the EDX API can be used to implement a custom browser specialized to a particular data source. In this case, the custom browser allows the user to navigate through the structured category information provided by the EIA using an interface that presents the categories as nested folders.

Requirements

The sample project is built using Visual Studio 2012. The project is currently set to target .NET Framework 3.5. If you change the project to target .NET Framework 4.0 or later you will need to replace the reference to EViewsEdx. `IDatabaseBrowserEvents` in `CategoryBrowserControl.vb` to EViewsEdxNet. `IDatabaseBrowserEvents`.

The project references the EViews Database Extension Type Library which is installed as part of EViews 8.1. Note that use of the EDX API to access custom data sources is only supported by the Enterprise edition of EViews.

The distribution also includes example installation projects implemented using InstallShield Limited Edition. You will not be able to open these projects unless InstallShield is available on your system. InstallShield Limited Edition is free for users of all Visual Studio editions except the Express editions, but must be downloaded and installed separately. To enable InstallShield Limited Edition choose File, New, Project from within Visual Studio and then navigate to **Setup and Deployment** and find the item **Enable InstallShield Limited Edition**. Follow the instructions to enable the product.

Solution Files

There are two Visual Studio 2012 solution files included in this distribution.

EdxEia.sln	This solution contains only the Visual Basic .NET project. Use this solution if you do not have InstallShield available on your system.
EdxEiaWithInstallers.sln	This solution file includes the Visual Basic .NET project as well two InstallShield Limited Edition projects (a 32 bit and a 64 bit version) that demonstrate how to distribute a database extension so that it is correctly installed onto a user's system. You must have InstallShield available on your system in order to open this solution.

Project Files

The distribution contains the following projects:

EdxEia.vbproj	The visual basic project that implements the database extension
EdxEiaSetup.isproj	An InstallShield Limited Edition project that installs the database extension on systems running 32-bit EViews
EdxEiaSetup64.isproj	An InstallShield Limited Edition project that installs the database extension on systems running 64-bit EViews.

Code Files

The main Visual Basic .NET project contains the following files:

EiaServerDbMgr.vb	The database manager class that supports online access to EIA data. One database manager is constructed per EViews session.
EiaServerDb.vb	The database class responsible for online access to the EIA web server. An instance of this database class is constructed each time the EIA online database is opened within EViews.
EiaFileDbMgr.vb	The database manager supporting EIA bulk file format. One database manager is constructed per EViews session.
EiaFileDb.vb	The database class for offline access to EIA data via the EIA bulk download file format. An instance of this database class is constructed each time a user opens an EIA bulk file inside EViews.
EiaParser.vb	A parser class that understands JSON content provided by the EIA. The parser class is also responsible for caching category information that has been returned by the server or that was seen during the initial scan of a bulk download file.
CategoryBrowserControl.vb	A custom browser control that allows navigation through the EIA category information. The custom browser is constructed by EViews whenever a user clicks on the 'Browse' button in the database window toolbar.
DownloadDialog.vb	A simple dialog that walks the user through the download of an EIA bulk file.
ServerPreferencesDialog.vb	A simple dialog that lets the user enter the EIA API key information they obtained when they registered with the EIA.

Customizing the Project

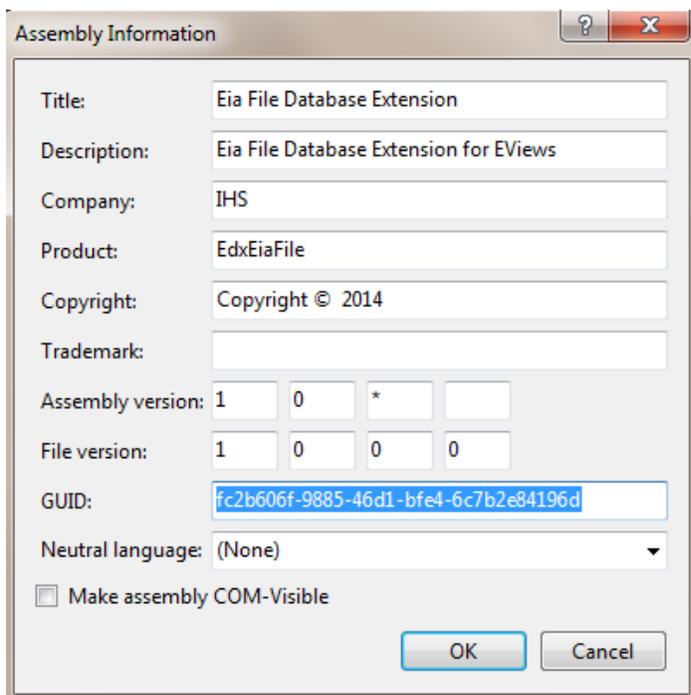
This sample contains several identifiers that are used by EViews and must be changed if you are interested in using the example project as a starting point for your own development. Failing to change these identifiers may cause the EIA database extension or your own extension to fail to operate correctly on a system on which your database extension is installed.

The most important identifier is the ProgId of each Database Manager class. Here is the code containing the ProgId from the top of the ServerDatabaseManager class.

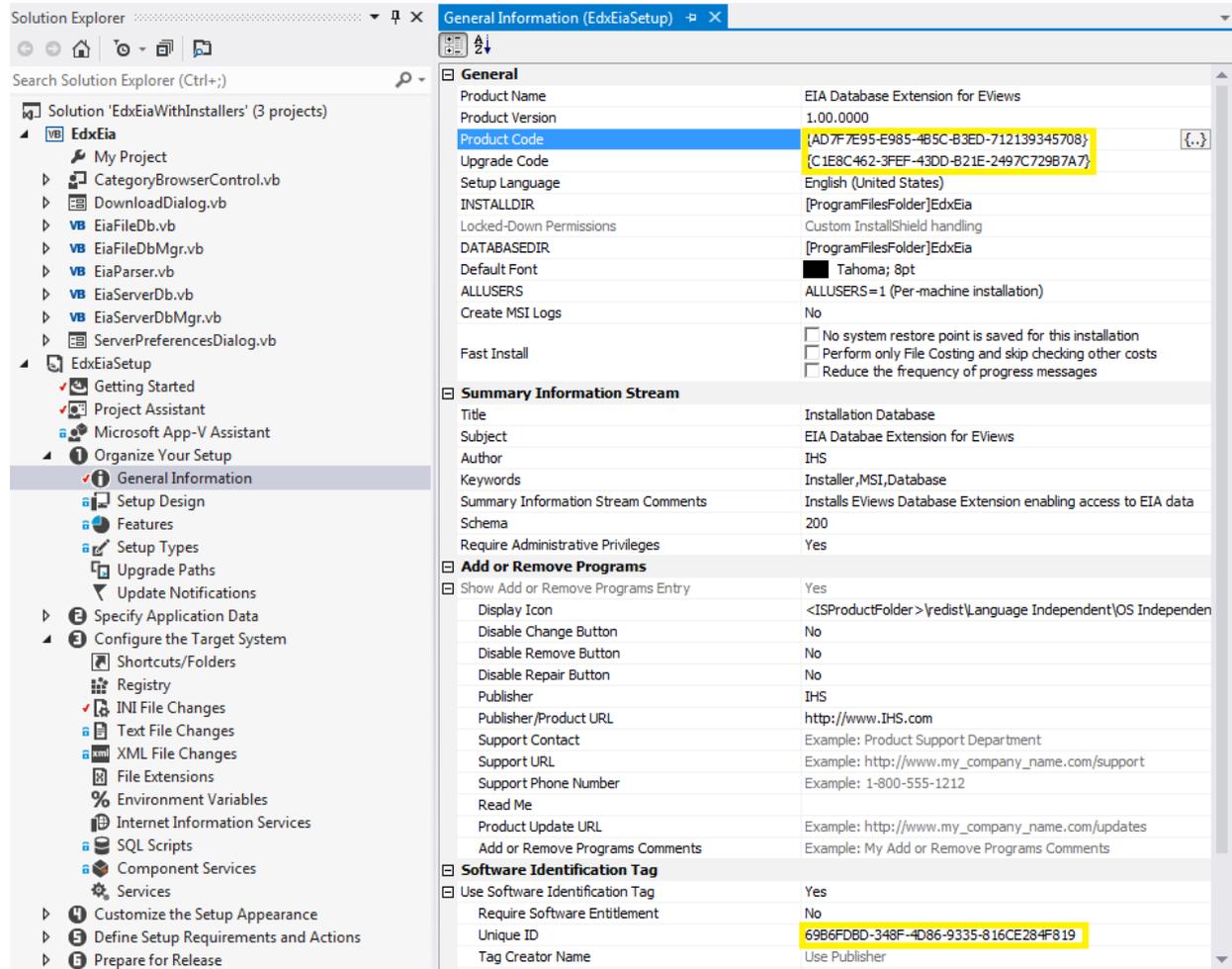
```
<ProgId(ServerDatabaseManager.myProgId),  
    ClassInterface(ClassInterfaceType.None), _  
    ComVisible(True)> _  
Public Class ServerDatabaseManager  
    Implements EViewsEdx.IDatabaseManager  
  
    'data members of class  
    Public Const myProgId As String = "EdxEia.ServerDatabaseManager"
```

You must change the myProgId string to a different value if you are customizing the project. This string is used by EViews to identify the database manager and failing to change this will make it impossible for EViews to tell your own database extension apart from the EIA extension.

You should also change the assembly information (available under Project Properties) to use a different Product Name and GUID. (Use Tools...Create GUID... from the Visual Studio menus to generate a new GUID value).



If you are using the InstallShield Limited Edition example projects you should also be careful to change the identifiers for your own installers to unique values so they do not interfere with other installers. You can find GUIDS for Product Code, Upgrade Code and Unique ID under Organize Your Setup, General Information. Click on the button labeled {...} next to each value to generate a new GUID.



Further material

Please consult the main EDX API documentation for a discussion of the purpose and usage of each function in the IDatabaseManager and IDatabase interfaces.