using uigen.exe

As the PhoneBook tutorial explains, `uigen.exe` is a fairly robust tool as soon as you got the configuration and the setup right. `uigen.exe` is picky in that

- It insists on absolute paths in the uigen.exe configuration file. A consequence is that if `uigen.exe` works for your project, you have to edit the `uigen.exe` configuration file to make it work for a different directory. This can bite you if you don't know this and try to duplicate a successful generation for a web project on a different disk or in a different directory.
- The discovery mechanism for domain object classes is much less sophisticated than that for dbschema.exe. You have to get everything right to the dot, or
  - `uigen.exe` will complain that it can't load the domain assembly(s)
  - the generated project will contain unresolved references
  - you will have problems using BOC controls in Visual Studio's designer
  - all of the above.

The following instruction assumes the canonical project structure, as demonstrated with the PhoneBook project structure. We also assume that the re-motion assemblys are located in a sub-directory of the project root (as explained in in PhoneBook project structure).

`uigen.exe` data comes from several sources:

- the domain assembly/ies
- the configuration file
- (the generator master file `TabbedEditor.xml`, but this is under the hood for the purpose of common `uigen.exe` tasks.)

With all this considered, the invocation for the PhoneBook project looks like this:

```
PhoneBook.Domain\bin\Debug\uigen.exe /uigen:PhoneBook.xml
/asmdir:Phonebook.Domain\bin\Debug
```

Before you can invoke it, however, some preparations are required.

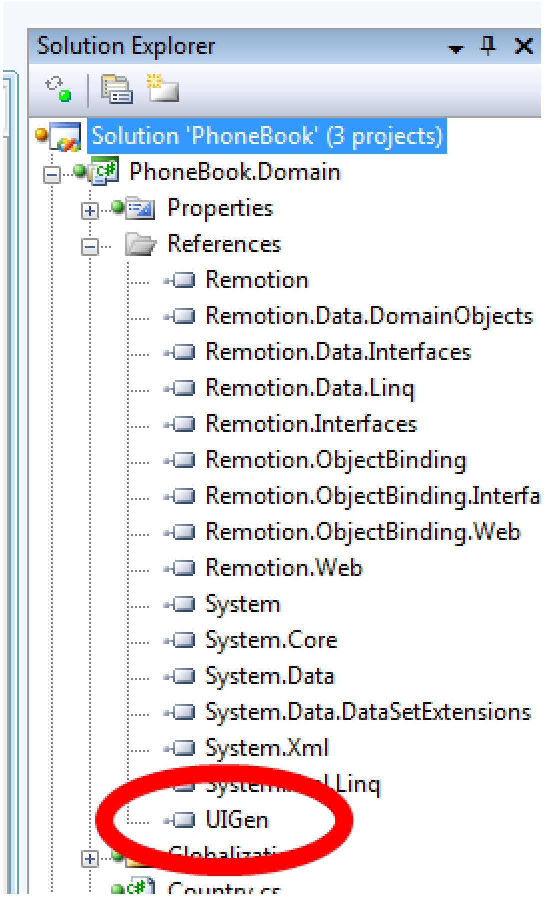
Making sure that uigen.exe finds all assemblies

`uigen.exe` must load re-motion assemblys for reasons:

- for discovering domain object classes – the basis for your web application
- for doing its work – because some re-motion assemblies contain code `uigen.exe` needs

The simplest trick is to leave all the hard work to Visual Studio and do the following:

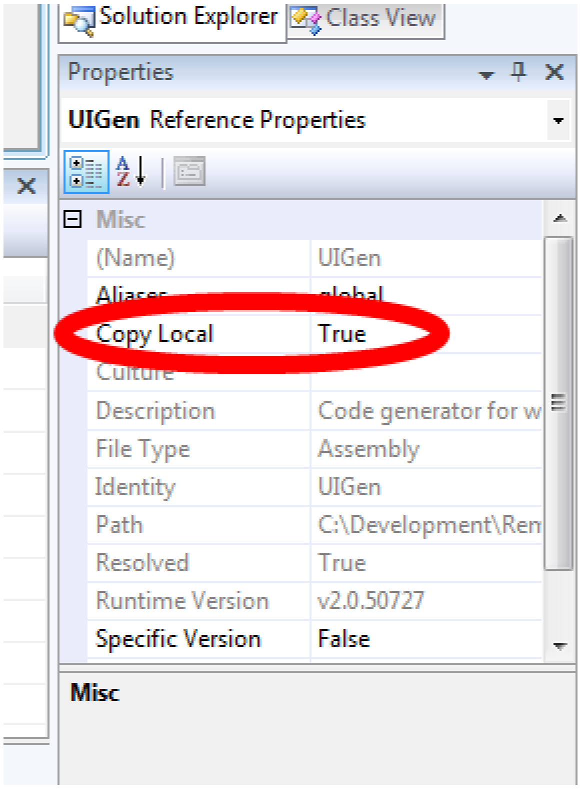
- include `uigen.exe` in your **domain project**, i.e. make the `uigen.exe` executable part of the ensemble of referenced re-motion assemblys. For the `PhoneBook.Domain` project this looks like this:



Solution Explorer

```
Solution 'PhoneBook' (3 projects)
  PhoneBook.Domain
    Properties
    References
      Remotion
      Remotion.Data.DomainObjects
      Remotion.Data.Interfaces
      Remotion.Data.Linq
      Remotion.Interfaces
      Remotion.ObjectBinding
      Remotion.ObjectBinding.Interfa
      Remotion.ObjectBinding.Web
      Remotion.Web
      System
      System.Core
      System.Data
      System.Data.DataSetExtensions
      System.Xml
      System.Xml.Linq
      UIGen
    Globalizati
    Country.cs
```

Make sure you have "Copy Local" set to TRUE, otherwise Visual Studio won't fix the dependencies for you:



UIGen Reference Properties

Misc

(Name)	UIGen
Aliases	global
Copy Local	True
Culture	
Description	Code generator for w
File Type	Assembly
Identity	UIGen
Path	C:\Development\Rem
Resolved	True
Runtime Version	v2.0.50727
Specific Version	False

Misc

DONT FORGET TO REBUILD YOUR REBUILD YOUR DOMAIN PROJECT

Configure

The uigen.exe configuration file is an XML file and contains mainly "placeholders" for expanding all the project templates. Not all of them are relevant for your project, some are "legacy placeholders". The configuration file is an XML file and has two major nodes:

- `<settings>` – paths needed by `uigen.exe` to find its template files and resources
- `<placeholders>` – that's your configuration

A complete discussion of the configuration file can be found here: uigen.exe configuration file. A sample configuration file can be found here: sample uigen.exe configuration file.

The canonical place for the `uigen.exe` configuration file is the solution root, i.e. above all your Visual Studio sub-projects. By convention, it has the same name as the entire project. For the PhoneBook project that's `PhoneBook.xml`.

Run `uigen.exe`

For the PhoneBook project, the invocation looks like this. In the solution root, invoke

```
PhoneBook.Domain\bin\debug\uigen.exe /uigen:phonebook.xml
/asmdir:phonebook.domain\bin\debug
```

Again, this assumes that

- the `uigen.exe` executable has been copied to the domain project's `bin\Debug` (or `bin\Release` directory by Visual Studio (as explained in
Making sure that uigen.exe finds all assemblies)
- your current directory is the solution root (so that the domain project is located in a sub-directory)

- The `/uigen:` switch specifies the name of the `uigen.exe` configuration file.
- The `/asmdir:` switch specifies the directory where the domain assemblys can be found. (This is inconsistent with `dbschema.exe`'s `/baseDirectory:` switch, but it has the same meaning.)