

Microsoft Visual Studio Project Type GUIDs and Project GUIDs

Daniel Fowler, Eye Internet Limited

A **G**lobally **U**nique **I**dentifier (GUID) is a 128 bit integer, storable in 16 bytes, that is used whenever a number is required to identify something within a computer system, usually software. It can be used to identify pieces of code, database rows, data transmissions, software components, programs, and even hardware or parts of hardware. The number is generated when it is first required and then is never (usually) changed. There is no predefined list of identifiers. The algorithm used to generate the number ensures an extremely low probability of it ever being duplicated in other systems or on the same system, since there are 2^{128} (3.4×10^{38}) different values.

The identifier is normally displayed in hexadecimal to allow for concise and manageable handling. The format being four bytes hex, hyphen, followed by three lots of two byte hex separated by hyphens, then another hyphen and then six bytes in hex. This format is related to the algorithm that produces the number, here is an example of a GUID:

```
668D9867-B455-48CC-A7FA-AB2D9D9E9BDC
```

The Microsoft program GUIDGEN.EXE can be used to generate a new identifier when one is required, or in C# use "System.Guid.NewGuid()".

Microsoft Visual Studio uses GUIDs to identify a Project Type (C#, Visual Basic, Web, etc.) and to uniquely identify an individual project, a particular DLL or EXE. Thus a C# EXE would have a different Project Type GUID to a Visual Basic DLL, but another C# EXE would have the same Project Type GUID. However, two C# EXE projects would have different project GUIDs.

So for a C# EXE project you will find in the solution file (.sln):

```
...  
Project ( "{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}" ) =  
"WindowsFormsTestApp",  
"WindowsFormsTestApp\WindowsFormsTestApp.csproj", "{F00515D6-939F-  
4E92-8FFD-A5D8F811FF84}"  
...
```

Here "FAE04EC0-301F-11D3-BF4B-00C04F79EFBC" signifies a C# Windows Exe project. And that particular EXE project which is uniquely identified as "F00515D6-939F-4E92-8FFD-A5D8F811FF84". And in the project file (.csproj) you'll find that the second GUID will match:

```
<ProjectGuid>{F00515D6-939F-4E92-8FFD-A5D8F811FF84}</ProjectGuid>
```

This allows for developers in different companies to call projects the same name but if they come across each other's work the same named projects would still be uniquely identifiable by Visual Studio.

Adding another C# EXE project to the solution produces the following in the solution file (.sln):

```
...  
Project ( "{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}" ) =  
"WindowsFormsTestApp" ,  
"WindowsFormsTestApp\WindowsFormsTestApp.csproj" , "{F00515D6-939F-  
4E92-8FFD-A5D8F811FF84}"  
...  
Project ( "{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}" ) =  
"WindowsFormsApp2" , "WindowsFormsApp2\WindowsFormsApp2.csproj" ,  
"{F156E46F-FFA7-445E-B917-D1E47A6CEDAB}"  
...
```

Thus the Project Type GUIDs are the same, signifying the same type of project, but the project GUIDs are unique, indicating that they are not the same project (as obviously indicated by their names).

If a project is ever copied, usually by copying a directory and all its files, and then that project is changed, becoming a new project in its own right, then to keep Visual Studio happy you should change the project GUID. Use GUIDGEN.EXE to generate a new GUID. Open the copied project file (.csproj) and replace the project GUID with the new one. Then open the copied solution file (.sln) and replace all occurrences of the original GUID with the new one.

You will also come across GUIDs in the source files as well, usually when a .NET Assembly (EXE or DLL) can be used with legacy applications as a COM component or application, or when a .NET application needs to use a legacy COM component or ActiveX DLL. All these GUIDs are in no way linked to the GUIDs in the solution and project files. The GUIDs used for COM interoperation are attached to .NET assemblies using .NET attributes, and MUST be different values from the GUIDs in the solution and project files. Likewise other uses of GUIDs exist as identifiers for other elements in a program or database. Again they are not linked to the Project Type GUIDs or project GUIDs in anyway (unless a project is a Visual Studio Add-In or manipulates Visual Studio solution and project files).

Project Type GUIDS will not normally need to change if a project is copied (unless the project is being targeted at a completely different environment, in which case many other types of changes will also need to be performed). However, tools and Visual Studio Add-Ins

may need to know about Project Types. Those interested in the different Project Type GUIDs can refer to the following table:

Project Type Description	Project Type Guid
Windows (C#)	{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}
Windows (VB.NET)	{F184B08F-C81C-45F6-A57F-5ABD9991F28F}
Windows (Visual C++)	{8BC9CEB8-8B4A-11D0-8D11-00A0C91BC942}
Web Application	{349C5851-65DF-11DA-9384-00065B846F21}
Web Site	{E24C65DC-7377-472B-9ABA-BC803B73C61A}
Distributed System	{F135691A-BF7E-435D-8960-F99683D2D49C}
Windows Communication Foundation (WCF)	{3D9AD99F-2412-4246-B90B-4EAA41C64699}
Windows Presentation Foundation (WPF)	{60DC8134-EBA5-43B8-BCC9-BB4BC16C2548}
Visual Database Tools	{C252FEB5-A946-4202-B1D4-9916A0590387}
Database	{A9ACE9BB-CECE-4E62-9AA4-C7E7C5BD2124}
Database (other project types)	{4F174C21-8C12-11D0-8340-0000F80270F8}
Test	{3AC096D0-A1C2-E12C-1390-A8335801FDAB}
Legacy (2003) Smart Device (C#)	{20D4826A-C6FA-45DB-90F4-C717570B9F32}
Legacy (2003) Smart Device (VB.NET)	{CB4CE8C6-1BDB-4DC7-A4D3-65A1999772F8}
Smart Device (C#)	{4D628B5B-2FBC-4AA6-8C16-197242AEB884}
Smart Device (VB.NET)	{68B1623D-7FB9-47D8-8664-7ECEA3297D4F}
Solution Folder	{66A26720-8FB5-11D2-AA7E-00C04F688DDE}
Workflow (C#)	{14822709-B5A1-4724-98CA-57A101D1B079}
Workflow (VB.NET)	{D59BE175-2ED0-4C54-BE3D-CDAA9F3214C8}
Deployment Merge Module	{06A35CCD-C46D-44D5-987B-CF40FF872267}
Deployment Cab	{3EA9E505-35AC-4774-B492-AD1749C4943A}
Deployment Setup	{978C614F-708E-4E1A-B201-565925725DBA}
Deployment Smart Device Cab	{AB322303-2255-48EF-A496-5904EB18DA55}
Visual Studio Tools for Applications (VSTA)	{A860303F-1F3F-4691-B57E-529FC101A107}
Visual Studio Tools for Office (VSTO)	{BAA0C2D2-18E2-41B9-852F-F413020CAA33}
Visual J#	{E6FDF86B-F3D1-11D4-8576-0002A516ECE8}
SharePoint Workflow	{F8810EC1-6754-47FC-A15F-DFABD2E3FA90}
XNA (Windows)	{6D335F3A-9D43-41b4-9D22-F6F17C4BE596}
XNA (XBox)	{2DF5C3F4-5A5F-47a9-8E94-23B4456F55E2}
XNA (Zune)	{D399B71A-8929-442a-A9AC-8BEC78BB2433}
SharePoint (VB.NET)	{EC05E597-79D4-47f3-ADA0-324C4F7C7484}
SharePoint (C#)	{593B0543-81F6-4436-BA1E-4747859CAAE2}
Silverlight	{A1591282-1198-4647-A2B1-27E5FF5F6F3B}