Shared Assemblies

Source	http://www.dotnetspider.com/kb/Article2154.aspx

Introduction

When we maintain one copy of the Assembly for set of applications through out the system it is called Shared Assembly
More memory usage can be reduced by this concept
In order to avoid the conflict of the version Shared Assemblies have been introduced uniqueness of Assembly can be provided with creating/generating public key.These keys will later be used to give our assembly the guaranteed unique name.
There are 3 steps

1.   Generation of Unique key
2.   Creating Assembly signing with public key
3.   Place the Assembly in GAC

Generation of Unique key

This key will later be used to give our assembly the guaranteed unique name
MS is providing a feature called sn.exe to generate a unique key
Sn refers to Strong Name .Running the Strong Name utility provides with a file containing the public and private keys to use

```
C:\>sn -k mykeypair.snk

Key Pair return to mykeypair.snk

Mykeyfile.snk will contain public key
```

Creating Assembly signing with public key

```
Example:  Go to Start-->Programs-->MicrosoftVisualStudio2003-->File-->NewProject--->Class Library -->SAssembly

namespace SAssembly
{
public class MathC
{
private  int a,b;
private  int res;
public  void Accept( int A,int B)
{
a =A;
b= B;
}
public int Sum()
{
res = a+b;
return res;
```

```
}
}
}
```
Save as -->SAssembly.cs in the location E:Projects\SAssembly
Before compilation We need to sign in key file with a public key

Complier will read public key from key field sn –k
We mention the signing information in a special file called Assembly Info. Open the file from VS.NET solution explorer and change it to include the following line:
[assembly: AssemblyKeyFile("C:\\KeyPair.snk")]

Compile the Application
SAssembly.dll is produced
When an assembly signed with public key then its called as strongly named assembly.Combination of Assembly name public key with version number is called strong name
Placing the Assembly into GAC

There are two method available for this
Drag and Drop
Using GAcutil.exe

1.Drag and Drop

Windows Explorer ?Go to the location of the folder where Shared Assembly is located and drag and drop that in assembly folder

2.using the Tool GACUtil.exe

Go to Visual Studio Command Prompt
Syntax:
C: gacutil   -i   " Assembly path "

C:\>gacutil -i "E:Projects\SAssembly\Bin\Debug\SAssembly.dll"
              Microsoft (R) .NET Global Assembly Cache Utility. Version 1.1.4322.573
Copyright (C) Microsoft Corporation 1998-2002. All rights reserved.

Assembly successfully added to the cache

This will automatically place the assembly in GAC with out drag and drop
To Delete the Assembly from GAC
Go to VS Command Prompt gacutil –u Assembly Name
C:\WINNT\assembly\GAC> gacutil -u SAssembly

Now call this Assembly in Client Application

~ ~ ~ End of Article ~ ~ ~