**[初识NuGet - 概念， 安装和使用](http://www.cnblogs.com/Dlonghow/archive/2012/03/16/2399993.html)**

1. NuGet是什么？

NuGet is a Visual Studio 2010 extension that makes it easy to add, remove, and update libraries and tools in Visual Studio projects that use the .NET Framework. This topic lists documentation that will help you use NuGet packages and create your own.

NuGet 是一个Visual Studio的扩展。在使用Visual Studio开发基于.NET Framework的应用时，NuGet能够令你在项目中添加、移除和更新引用的工作变得更加快捷方便。

这就是说: 我们往往在开发项目里，会使用一些第三方的dll, 如：Newtonsoft.Json.dll (它为我们提供了进行序列化和反序列化解决方案， 很有用，也很稳定，还很有名气)。

但是，我们往往也面临随着项目升级，对这些引入的第三方插件进行升级的可能和必要。

比方说，原本的项目是vs2008 + C# 3.5 实现的，并引入了Newtonsoft.Json.dll 2.0版本， 现在对项目升级成了Vs 2010 + C# 4.0。

如果 你想或者必须将旧版本升级到最新版本才能工作或者仅是保持版本上的一致； 同时，Newtonsoft 提供Json.Net对C# 4.0 支持版本，即：Newtonsoft.Json.dll 4.0。

你就需要到官网或者通过Google 去寻找它的最新版本，下载，解压，然后替代等操作。 似乎看上去，也没多少复杂的事情， 也习以为常了。谁叫咱引用别人的东西，只能跟着走了，习惯了这样的更新，总比自己写更快吧。

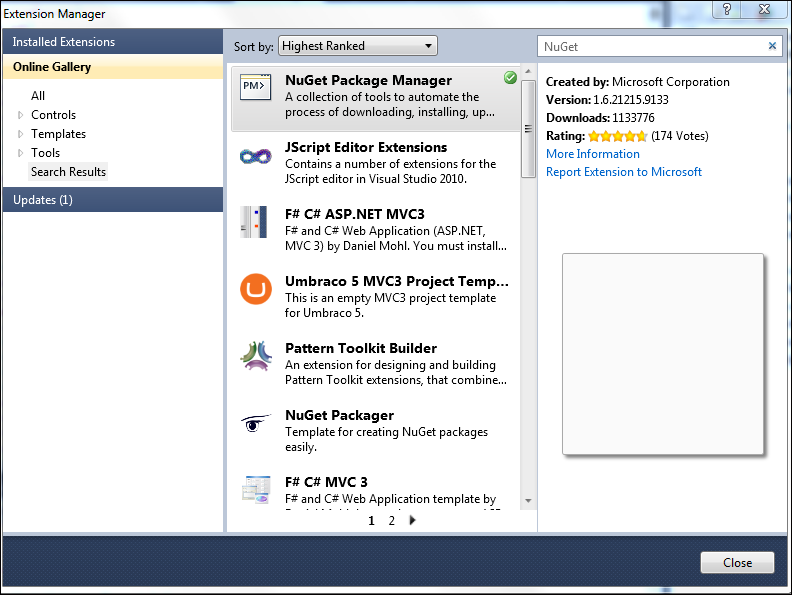
 现在，NuGet 将这个过程系统化，令你可以更加方便的查找你要的第三方的dll， 同时，还可以为你提供自动或者手动更新，而无需你在人肉搜索与更新了。

**2. 官方网站：www.nuget.org**

    CodePlex: http://nuget.codeplex.com/

**3. 安装：**

打开Visual Studio 2010, Tools-> Extensions Manager... （工具->扩展管理器）， 打开 vs扩展管理库，选择 Online Gallery (联机库)， 既可以在右上角搜索 “NuGet”。



选择“NuGet Packge Manager”, 点击右上角的 Install，既可以完成安装，安装完成后，即为 绿色的对号。

**4 . 举例：**

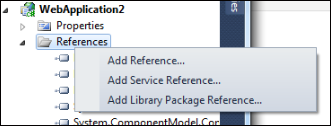
新建一个项目，这里我们就简单创建一个Web Application:

我打算在这个项目中，添加Newtonsoft.Json.dll。

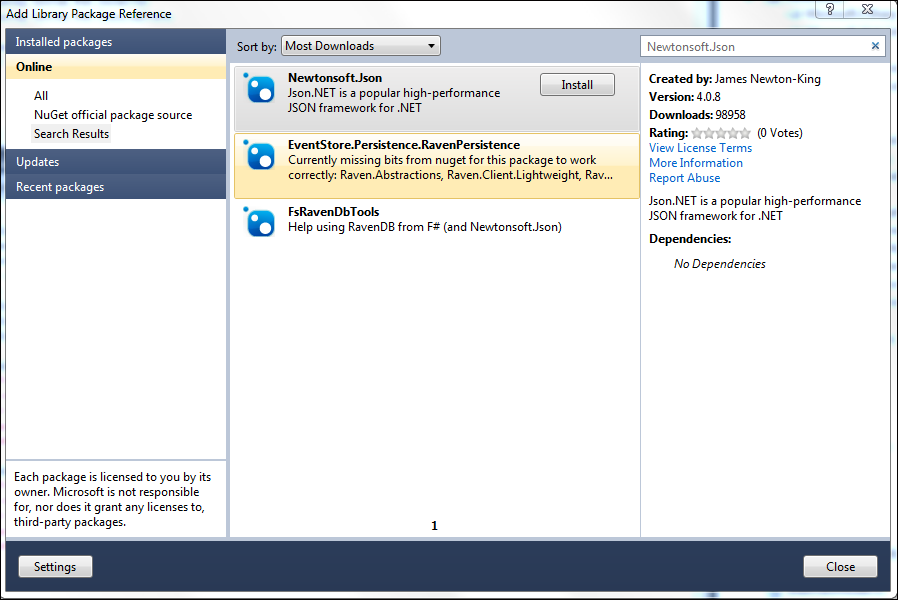
我不再通过传统方式，下载解压并copy到项目中再引用。而是通过NuGet提供的方式实现：

NuGet提供了两种方式添加（至少我值知道是两种方式）:

**第一种**通过 "Add Library Package Reference..." 添加，如下图所示：



点击 ‘Add Library Package Reference...’ , 搜索你要添加的Package的名称：

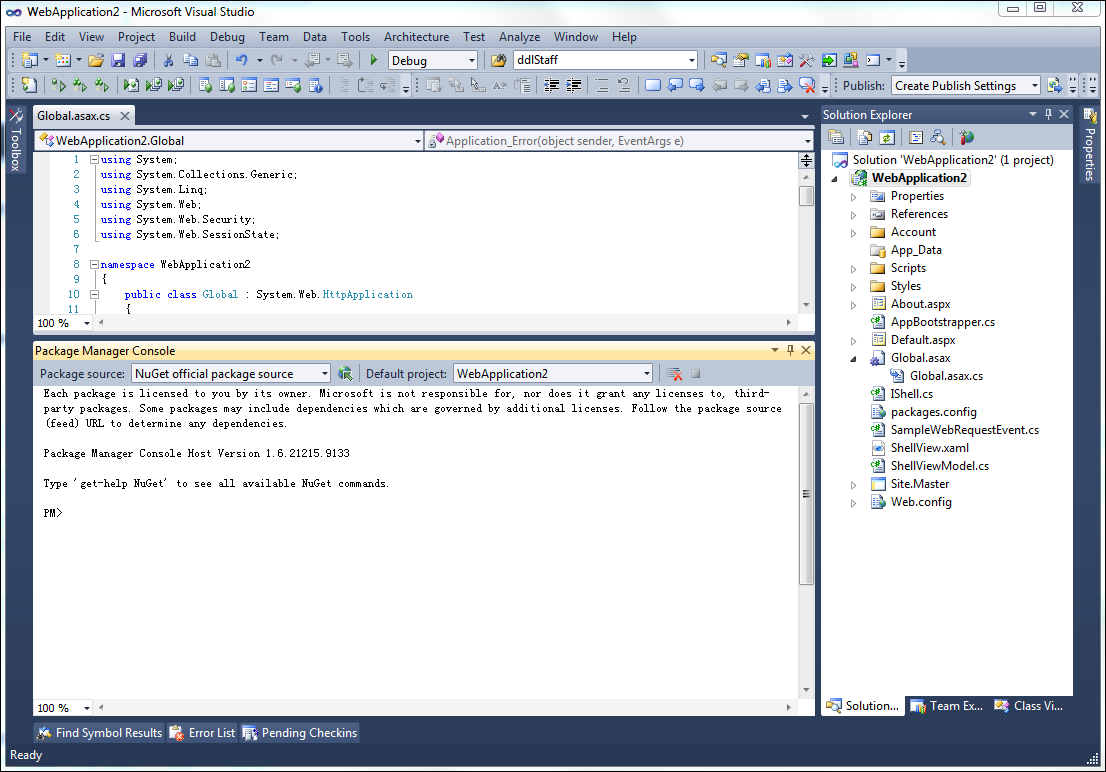


点击 "Install" 按钮， 安装这个Package。 这是你将发现NuGet为你当前的项目自动引入了最新的 Newtonsoft.Json.dll 动态链接库。

这时，你就可以使用它了。

第二种 使用Package Manager Console为项目添加该引用：

从VS2010的主菜单->Tools->Library Package Manager->Package Manager Console，打开Package Manager控制台：



假设我们知道 Newtonsoft.Json 在NuGet上的名称， 我们可以使用下面命令安装引用：

Install-Package Newtonsoft.Json

回车后，NuGet将自动为你下载并添加引用。 同时，控制台为你提示如下信息：

 PM> Install-Package Newtonsoft.Json

Successfully installed 'Newtonsoft.Json 4.0.8'.

Successfully added 'Newtonsoft.Json 4.0.8' to WebApplication2.

你还可以通过这两种 方式更新引用为最新版本。

**5. 创建自己的 NuGet Server。**

如果你是一个大公司的程序设计人员，你设计了解决某一问题的 解决方案即一个良好的会被同事频繁使用的动态链接库中NuGet.Demo.dll，你将它放置在公司共享的某个磁盘文件夹下，并email通知大家最新的更新等。

这个dll一直有你维护和更新，以往大家也就会得到你的email通知, 根据自己的需求，到上文说的文件夹下copy并删除当前旧的dll.

如今我们可以通过NuGet Server提供的手段，帮我们更加自动和人性化的简化并更加方便这一工作场景：

那么，如何创建这个自己公司的NuGet Server。

我们新创建一个 工程：Infoshare.Projects.

然后通过上面的方法，打开Package Manager Console, 并输入如下命令并回车：

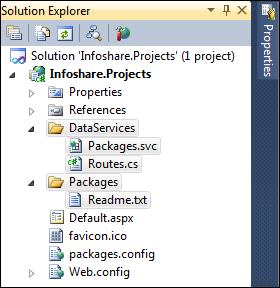
......

**PM> Install-Package NuGet.Server**

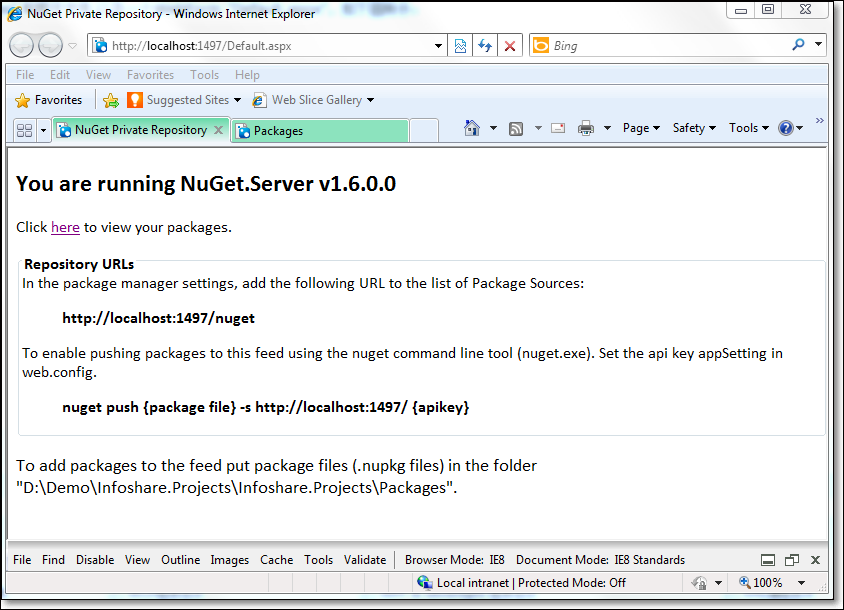
......

这时，你会发现工程的Refernces 下添加了一个NuGet.Core的引用，它将为我们实现这一方案。

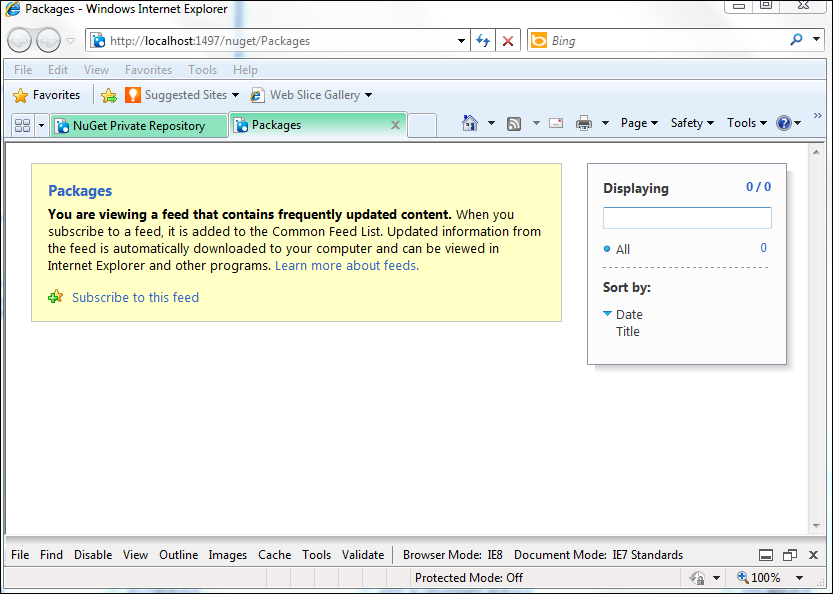
同时，添加了两个文件夹： “DataServices”  ；“Packages” 和若干文件 以及一个WebForm "Default.aspx"，如下图所示：



运行，并查看Default.aspx页面：



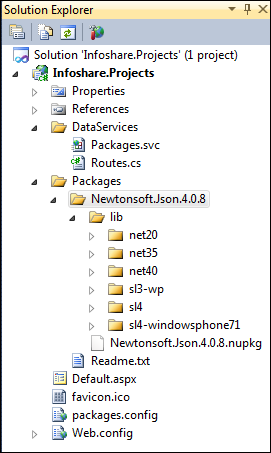
点击here，进入Packages页面， 你应该可以看到Package下包含的nuget包:



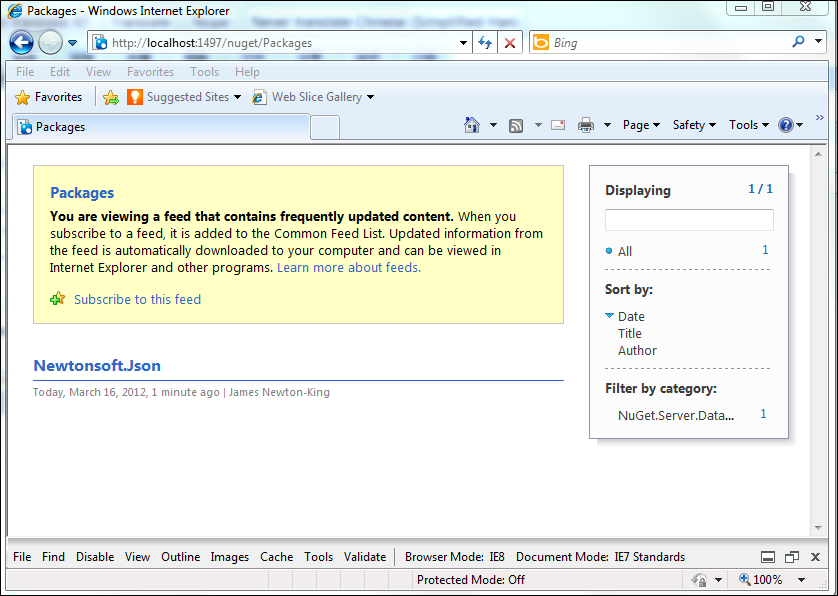
但是，如上面显示并没有任何Feed 记录。这是因为在刚才的Infoshare.Projects的Packages文件夹下，并没有任何nuget包。

我们将上面通过NuGet的Packages Manager Cnsole添加的 NuGet 包，放到当前这个Infoshare.Projects的Packages文件夹下，

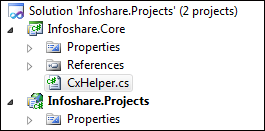
添加完成后如下图：



在运行查看：

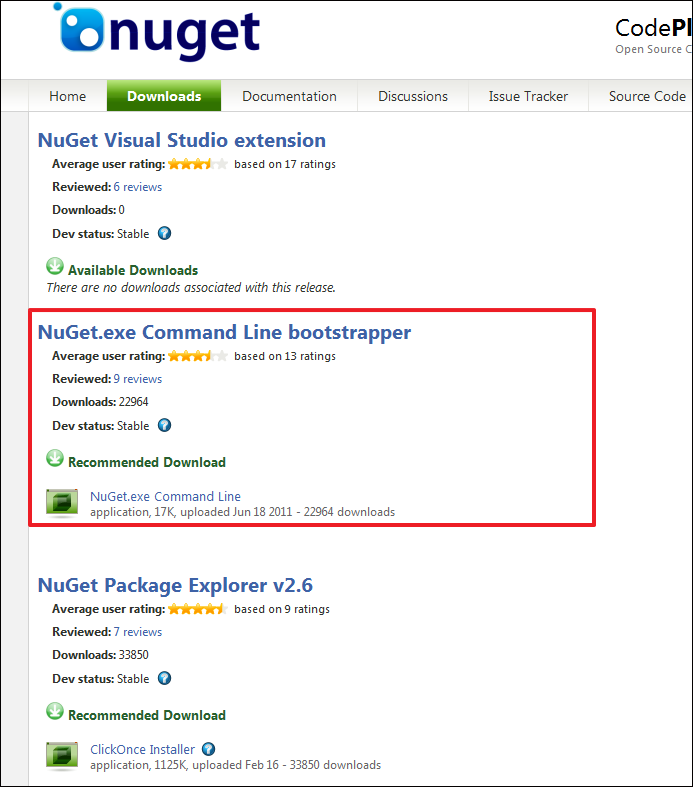


我们在项目中创建一个Class Library 命名为：**Infoshare.Core**, 将作为公司普用的关键dll，通过NuGet Server进行管理。



现在的问题是如何才能生成类似Newtonsoft.Json可供NuGet.Server使用的nuget包，即将自己的公司的dll打包成nuget包？

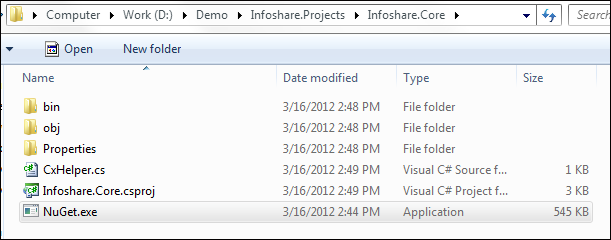
 这要用到NuGet提供的工具，你需要到 <http://nuget.codeplex.com/>上面下载：



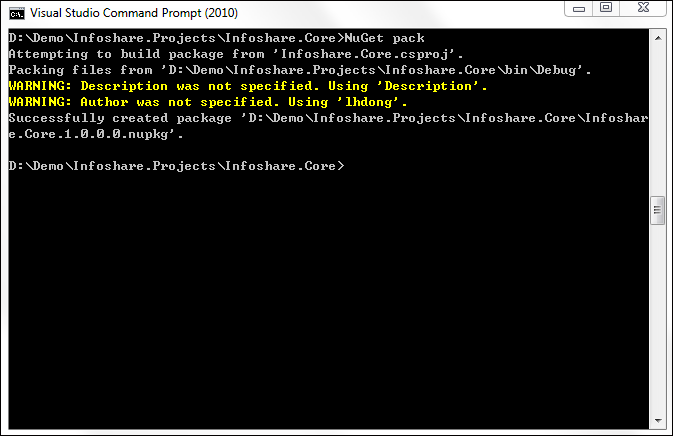
下载nuget.exe 后，你可以使用下面的方式进行打包：

 使用nuget.exe pack将工程文件打包， （命令："nuget pack"）：

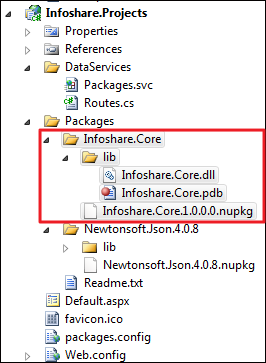
 将下载的NuGet.exe放到Infoshare.Core的工程文件夹下：



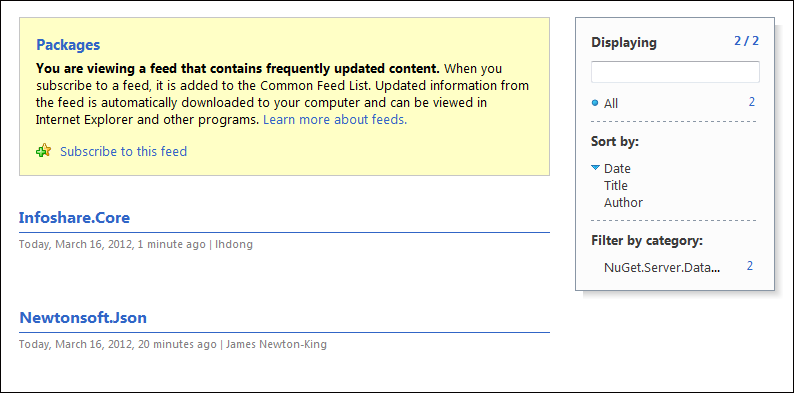
执行如下命令：



将生成的 **Infoshare.Core.1.0.0.0.nupkg**文件和相应的Infoshare.Core.dll 拷贝到 Infoshare.Projects下的Packages文件夹下：



在运行查看defalut.aspx:



那么如何像NuGet Server那样，可以通过“Add Library Package Reference...”，弹出Library Reference 搜索和安装对话框？

这就要向 NuGet 添加更多程序包源， 即添加你的NuGet Server程序包源。

通过“Tools”（工具）|“Library Package Manager”（库程序包管理器）|“Package Manager Settings”（程序包管理器设置）菜单选项，单击“Package Sources”（程序包源）节点，

