# Toolkit5.5下的源码调试

## Nuget

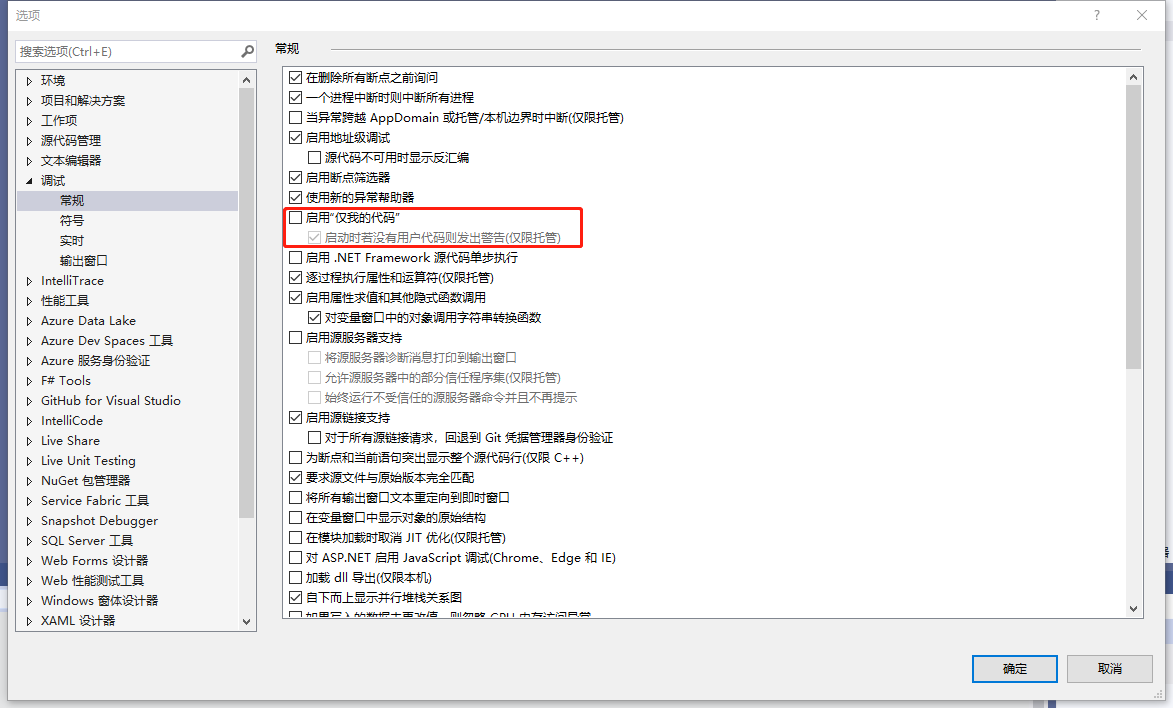
类似于Java平台上的Maven，Visual Studio上采用Nuget进行对引用的DLL进行管理。有了Nuget后，在任何机器上打开项目，就不用考虑项目的引用路径问题，Visual Studio根据需要自动下载相应的DLL。在无Visual Studio的环境下，通过运行dotnet restore亦可以达到相同效果。

## 如何进行源码调试

由于Nuget是从服务器上下载DLL，因此它不依赖本地是否有相应的项目。在发布Nuget包时，如果同时pdb符号文件和相关的源码，在Visual Studio下同样可以相关的源码调试。当然默认的开发环境是不行的，我们需要改变一些参数。

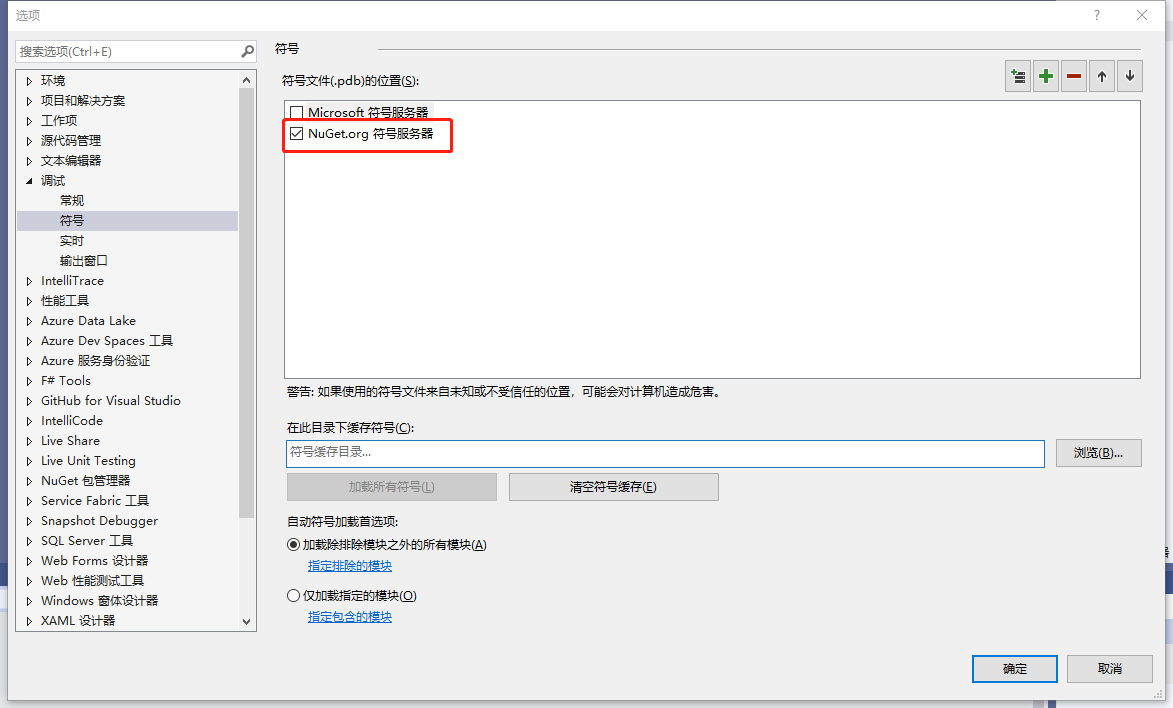
以Visual Studio2019（强烈推荐此版本）为例，我们需要做如下参数改变：

点击菜单“工具|选项”，在弹出的对话框中，选择调试|常规，如下图：



将“启用‘仅我的代码’”的选项点掉。默认是勾选上的。

再选择调试|符号，如下图：



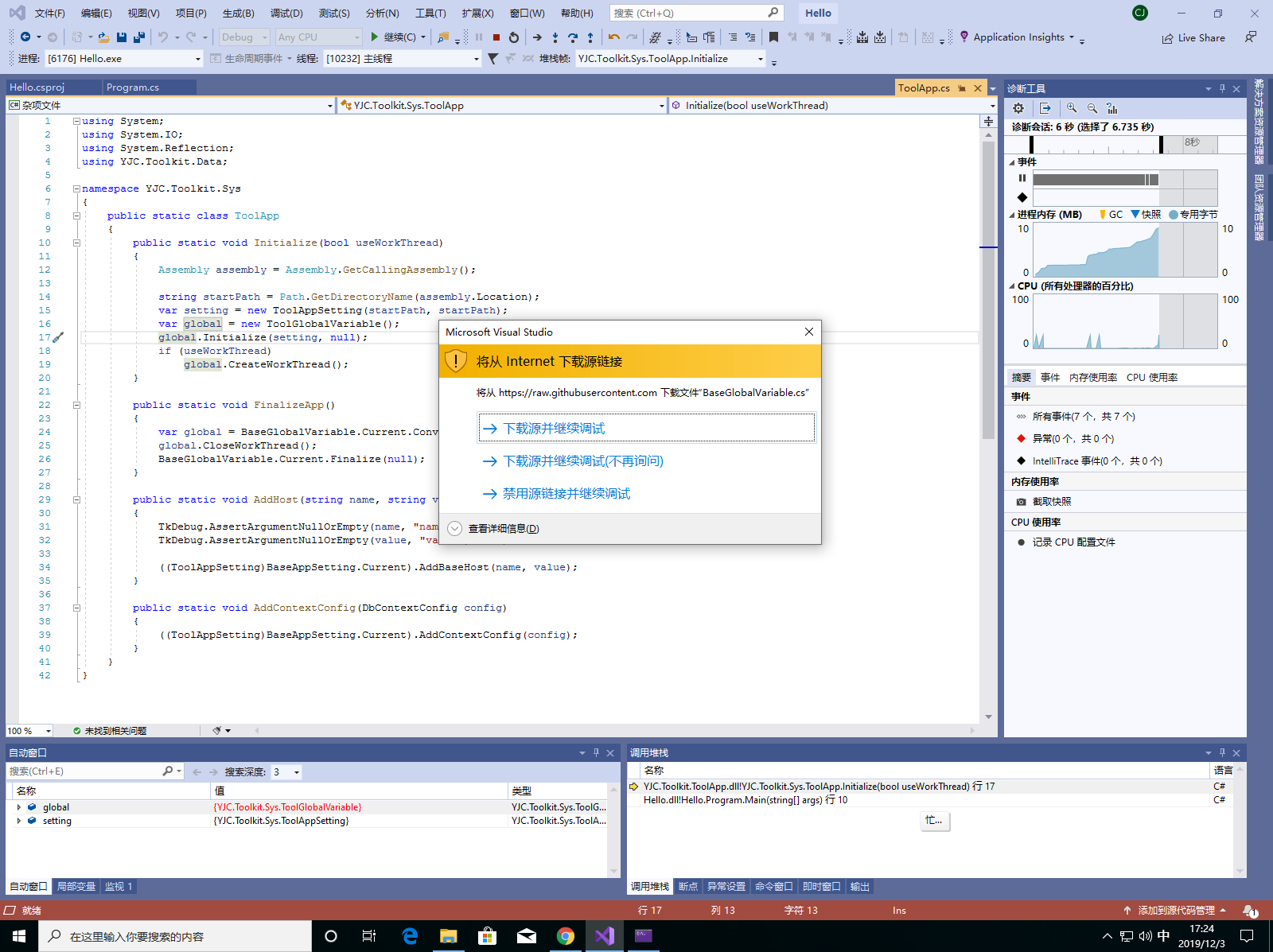
将NuGet.org符号服务器选中，默认是不选中的。

这里，不建议选择Microsoft符号服务器。如果选择了，那么启动的时候，它会去微软服务器下载.net系统的符号文件和源码，这会浪费大量的时间。当然，有兴趣调试微软的.net源码的，可以勾上该选项。当然第一次启动的时候，要有足够的耐心进行等待。

上述两个步骤后，就可以了。

下面和正常调试时，没有区别。当然第一次启动会比较慢，请耐心等待（如果选择了Microsoft符号服务器会更慢）。

当在Toolkit的代码上按F11时，如果是第一次，可能会出现如下对话框，如图：



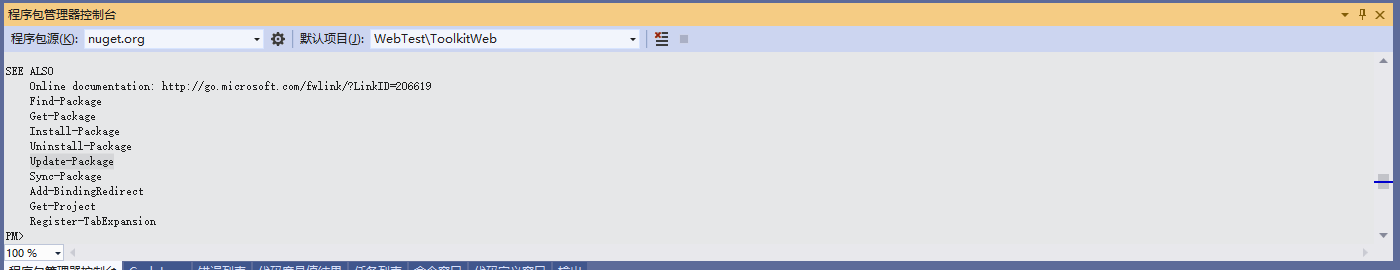
需要选择第一个或者第二个选项。然后就进入了Toolkit的源码。如果在调试Toolkit源码时，发现任何BUG，请及时反馈。

## 更新Nuget包

由于Toolkit系列的DLL都已经上传到Nuget的服务器上。当有相关功能更新或者BUG修复后，都会及时将最新的版本更新到Nuget服务器上。

那么现有的代码如何更新？这里提供一种比较便捷的方式。

选择菜单“工具|NuGet包管理器|程序包管理器控制台”，在Visual Studio下部的开发环境中，会出现一个类似于控制台的窗口，如下图：



在窗口中键入：

Update-Package

按回车，等候一会，当前solution中所有的项目的Nuget包都将会更新到最新版本。

（如果有Toolkit源码，建议不要做此操作。因为Toolkit的Razor的工程升级后，将无法编译通过，现阶段不打算解决上述问题。）