Visual Studio 2015和2017里面应该创建哪种 Class Library

2017-07-04 Lex Li [dotNET跨平台](https://mp.weixin.qq.com/s?__biz=MzAwNTMxMzg1MA==&mid=2654068833&idx=1&sn=89c7c283823c298ddb506ef39b90e89d&chksm=80dbc434b7ac4d223f530ff4410e2be55629d06c5baf444e01043fe4dddbf943b33ff56e9509&mpshare=1&scene=1&srcid=0704wZYtXsE8fI2Jcl9mOIMm&key=dc7dfb672b5ac5a814cea1aa275c7c163631cec53b9ad8d27a82d076d4290462feeb74fb552c7e2042ca9796aae2cfdcc8c9280cd853b9d47a119b62fee786b9c5db8b0c9bad0c0f83767c092a136cc8&ascene=1&uin=Mjg1Mzg5MzU%3D&devicetype=Windows+UnKnow&version=62040525&pass_ticket=Mq1890O9oNEOTT476IbpY5m3EHYRQR0JPGQ4sN%2BNGVk%3D&winzoom=1" \l "#)

微软有些时候是一副不讲道理的大家长作风。一个例子就是它在 Visual Studio 2015/2017 里面配置了一大堆普通人没法理解的 Class Library 模板。

你真的知道应该选哪个吗？

这就是 VS2017 里面你可能熟悉的长列表（而且因为我没有安装 UWP 开发项目，这个表还不完整）。其实仔细研究一下，也没有那么可怕了，它们实际上只有两类。

**平台相关的 Class Library**

从 .NET Framework 1.0开始，Class Library (.NET Framework）这个类型就存在了。所以闭着眼睛你也知道这种类型的工程只能被针对 .NET Framework 工程使用。有了这个基本认识，你同样也就知道 iOS、tvOS、watchOS、Android 和 UWP 等模板也是平台相关的，只有同一平台的工程才能够使用它。

虽然这样的模板生成的工程不能跨平台使用，但是它依然是一个合理划分项目粒度的好方式，使得每个 Class Library 内部高聚合，和其他工程之间低耦合。

**可以跨平台使用的 Class Library**

Class Library (Portable) 是微软最先想到的跨平台 Class Library 方案，一般叫做 PCL。由于它实际使用中的种种限制，微软现在已经不再推荐。

用来取代 PCL 的新方案，就是 Class Library (.NET Standard)。相比 PCL，这种类型的工程使得开发人员可以使用更大的 API 集合。也许你还需要一些实践来好好体会，不过现在简单背一下答案就好了，“**如果你需要开发跨平台的项目，这个模板就是你需要的**”。

当然，还有几个特例需要特别说明，.NET Core 和 Xamarin.Forms。

Class Library (.NET Core) 模板创建的工程只能被其他 .NET Core 项目使用。这当然和其他平台相关的 Class Library 模板一样。但是 .NET Core 是个特殊的平台，它可以运行在多个操作系统之上（这就出现了又一个跨平台）。.NET Core 的跨“平台”是指跨多个操作系统（Windows/Mac/Linux)。而 .NET Standard 的跨“平台”，是指它能够被多个平台（.NET Framework/.NET Core/Mono/Xamarin/Unity）的工程所引用。

Class Library (Xamarin.Forms) 也比较特殊。首先它暂时是基于 PCL 的模板，未来应该会升级为基于 .NET Standard 的模板。其次，能够引用它的工程，当前限制是 Xamarin 的相关工程，未来可能会增加其他工程类型。

**如何在这些工程之间迁移代码**

方式很多，不过最土的还是剪切粘贴了。你可以自己一一尝试。

本来嘛，这就是抛出的一块砖，希望没有打疼你，毕竟微软未来总会找到一个做文档的角度来把这些问题讲清楚吧。