# .Net4.5新功能

## 针对web

* 为新 HTML5 窗体类型的支持。
* 为模型联编程序支持在 Web 窗体中。 这些允许您直接将数据控件绑定到数据访问方法，并自动将用户输入转换到和来自 .NET Framework 数据类型。
* 为客户端验证脚本的不显眼的 [**JavaScript**](http://lib.csdn.net/base/18) 支持。
* 改进客户端脚本的处理通过改进页性能的绑定和缩减。
* 从 AntiXSS 库（以前的外部库）中集成编码例程可以从跨站点式脚本攻击中保护。
* 为 WebSockets 协议支持。
* 用于读取和写入 HTTP 请求和响应支持异步。
* 对于异步模块和处理程序。
* 为 ScriptManager 控件的内容分布式 Web (CDN) 回退支持。

## 新增的五个主要特性

1 异步和等待（代码标记）

关键字异步（async）和等待（await ）是用来标记代码在一个任务（线程）完成后，需要继续执行代码的所在位置的标记量。

2 ZIP附属功能（ZIP压缩）

在4.5中可以直接引用System.IO.Compression命名空间来完成文件压缩的一系列操作。

3 正则表达式超时

在4.5中可以为正则验证添加一个超时属性。所以，如果你收到任何类型的恶意注入时，你的应用不会陷入死循环。防止恶意的DOS攻击。

4 配置文件的优化

在.NET4.5中。我们有了一个叫做“配置文件优化”的一个利器。文件中只包含了程序启动时所需方法的列表。所以，当程序启动时，后台的JIT将会把这些方法翻译成机器语言/本地编码并运行。  
后台运行的JIT将跳过多数控制器，直接编译启动项配置文件里的方法，因此可以大大缩减所需时间。

5 垃圾回收器（GC后台处理）

在.NET4.0中，当GC清理垃圾时，所有应用程序的线程是被中止的。  
在某些事件节点上，后台的GC运行并开始清理。当这些后台的GC开始清理时，它们将中止所有应用程序的线程，这将导致在此时间节点上服务器和应用程序间的响应时间将会拉长。

为了解决以上问题，服务器端的GC应运而生。服务器GC会多开启一个后台运行的线程。这个线程将在后台运行，并且持续清理二代对象。因此，减少了主GC线程的负载。由于GC采用双线程运行，主程序的线程被中止的频率将大大境地。从而提高了应用程序的运行效率。