1、.aspx

Web窗体设计页面。Web窗体页有两部分组成：视觉元素（HTML、服务器控件和静态文件）和该页的编程逻辑（vs中的设计视图和代码视图可分别看到他们对应的文件）。Vs将这两个组成部分分别存储在一个单独的文件中。视觉元素在.aspx文件中创建。

2、.asax

Asp.net的用户控件，是作为一种封装了特定功能和行为（这两者要被用在web应用程序的各种页面上）的web页面被开发的。一个用户控件包含了HTML，代码和其他web或者用户控件的组合，并在web服务器上以自己的文件格式保存。

ASP.NET里的缺省配置并不允许web客户端通过URL来访问这些文件，但是这个网站的其他页面可以集成这些文件里包含的功能。

（asax是用户控件，用户控件必须嵌入到aspx中才能使用。）

3、ashx

（1）使用举例

.ashx文件是主要用来写web Handler的。使用.ashx可以让你专注于编程而不用管相关的web技术。我们熟知的.aspx是要做HTML控件树解析的，.aspx包含的所有HTML实际上是一个类，所有的HTML都是类里面的成员，这个过程在.ashx是不需要的。ashx必须包含IsReusable属性，（这个属性代表是否可复用，通常为true），而如果要在ashx文件用使用Session必须实现IRequiresSessionState接口。

分析：  
a、以前我们通常都是通过一个简单的aspx文件实现的功能，其实通过ashx也可以。  
笔者曾经写过的一篇[ajax:数据传输方式简介](http://www.cnblogs.com/jeffwongishandsome/archive/2009/01/10/1326012.html) ，通过对比，我们发现aspx要将前后台显示和处理逻辑分开,所以就弄成了两个文件,其实,在最终编译的时候,aspx和cs还是会编译到同一个类中去.这中间就要设计html的一些逻辑处理；而ashx不同,它只是简单的对web http请求的直接返回你想要返回的结果.比aspx少处理了html的过程（但是ashx也可以处理html的一些逻辑，只是通常都不这么用）。理论上ashx比aspx要快。  
b、还是在相同的旧文里，我们知道数据传输的几种方式，其实ashx都可以实现（修改ashx文件里context.Response.ContentType 即可），这里不再赘述了。  
（2）、ashx特别适合于生成动态图片，生成动态文本(纯文本，json，xml，javascript等即可)等。   
（3）、.ashx文件有个缺点：它**处理控件的回发事件**非常麻烦。处理数据的回发，通常都需要一些.aspx页的功能，只有自己手动处理这些功能（还不如直接建一个aspx文件来处理）。所以，一般使用.ashx输出一些不需要回发处理的项目即可。   
4、总结  
aspx-->P（Page）  
asax-->C（Control）  
ashx-->H（HttpHandler）

当浏览器访问web服务器，我们最终接收到的还是html文本。浏览器通过渲染引擎解释这些标签，在屏幕上展现出可见的效果。而asp.net不过就是我们应用的一种平台技术来"变相"解释html的，说白了它就是为了提高生产率，它的技术术语再多，本质上还是html范畴内的东西（如果你不通过那些动态页面技术完全利用html和浏览器（当然包括js）技术来实现动态页面效果，那么你会发现效果有了代码量也相当可观）.所以说web开发的底层就是一堆的html标签，无论是asp.net还是jsp都是对html某种方式的包装，是html的产物。

Ajax：传输方法简介

http://www.cnblogs.com/jeffwongishandsome/archive/2009/01/10/1326012.html