IIS 7.0 集成管道是一种统一的请求处理管道，它同时支持本机代码和托管代码模块。实现 [IHttpModule](https://msdn.microsoft.com/zh-cn/library/system.web.ihttpmodule(v=vs.100).aspx) 接口的托管代码模块可访问该请求管道中的所有事件。

结构概述

在 IIS 7.0 中，ASP.NET 运行时与 Web 服务器集成，这样就有了一个针对所有请求的统一的请求处理管道。对于 ASP.NET 开发人员，集成管道有以下益处：

集成管道引发由 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 对象公开的所有事件，这使现有 ASP.NET HTTP 模块可在 IIS 7.0 集成模式下工作。

在 Web 服务器级别、网站级别或 Web 应用程序级别，都可配置本机代码和托管代码模块。这包括用于会话状态、Forms 身份验证、配置文件以及角色管理的内置 ASP.NET 托管代码模块。此外，可以为所有请求启用或禁用托管代码模块，无论请求是否针对 ASP.NET 资源（如 aspx 文件）。

可以在管道中的任何阶段调用托管代码模块。这包括在对请求进行任何服务器处理之前，在所有服务器处理都已发生之后，或者两者间的任何阶段。

可以通过应用程序的 Web.config 文件注册模块，也可以启用或禁用模块。

下面的插图演示应用程序的请求管道的配置。该示例包括以下内容：

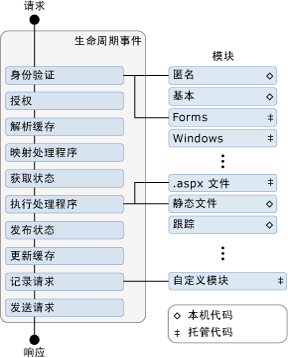
**Anonymous** 本机代码模块和 **Anonymous** 托管代码模块（对应于 [FormsAuthenticationModule](https://msdn.microsoft.com/zh-cn/library/system.web.security.formsauthenticationmodule(v=vs.100).aspx)）。这些模块都已配置，并在请求的**Authentication**阶段调用。

**Basic** 本机代码模块和 **Windows** 托管代码模块（对应于 [WindowsAuthenticationModule](https://msdn.microsoft.com/zh-cn/library/system.web.security.windowsauthenticationmodule(v=vs.100).aspx)）。它们虽然显示，但并没有为应用程序进行配置。

在 **Execute handler** 阶段，会调用处理程序（应用范围为 URL 的模块）以构造响应。对于 .aspx 文件，使用 [PageHandlerFactory](https://msdn.microsoft.com/zh-cn/library/system.web.ui.pagehandlerfactory(v=vs.100).aspx) 处理程序来响应请求。对于静态文件，本机代码 **StaticFileModule** 模块响应请求。

**Trace** 本机代码模块。此模块虽然显示，但并没有为应用程序进行配置。

**Custom module** 托管代码类。该类在 **Log request** 阶段中调用。



生命周期阶段

下表列出了在 IIS 7.0 集成模式下运行的 ASP.NET 应用程序生命周期的各个阶段。

|  |  |
| --- | --- |
| **阶段** | **说明** |
| 发出一个对应用程序资源的请求。 | 在 IIS 7.0 集成模式下，由一个统一的管道处理所有请求。当集成管道收到请求时，该请求将经历所有请求共有的一些阶段。这些阶段由 [RequestNotification](https://msdn.microsoft.com/zh-cn/library/system.web.requestnotification(v=vs.100).aspx) 枚举表示。所有请求都可以配置为使用 ASP.NET 功能，因为该功能封装在可以访问请求管道的托管代码模块中。例如，即使 .htm 文件扩展名未显式映射到 ASP.NET，对 HTML 页的请求仍会调用 ASP.NET 模块。这使您能对所有资源使用 ASP.NET 身份验证和授权。 |
| 统一管道接收对应用程序的第一个请求。 | 当统一管道接收对应用程序中的任何资源的第一个请求时，将为 [ApplicationManager](https://msdn.microsoft.com/zh-cn/library/system.web.hosting.applicationmanager(v=vs.100).aspx) 类创建一个实例，该实例就是处理请求的应用程序域。在应用程序域中，将为[HostingEnvironment](https://msdn.microsoft.com/zh-cn/library/system.web.hosting.hostingenvironment(v=vs.100).aspx) 类创建一个实例，该实例提供对有关应用程序的信息（如存储该应用程序的文件夹的名称）的访问。  在第一个请求期间，如果需要，将对应用程序中的顶级项进行编译，其中包括 App\_Code 文件夹中的应用程序代码。可以根据本主题后面的 [IIS 7.0 中的托管代码模块](https://msdn.microsoft.com/zh-cn/library/bb470252(v=vs.100).aspx" \l "Modules)中的说明，在 App\_Code 文件夹中包含自定义模块和处理程序。 |
| 将为每个请求创建响应对象。 | 在创建了应用程序域并对 [HostingEnvironment](https://msdn.microsoft.com/zh-cn/library/system.web.hosting.hostingenvironment(v=vs.100).aspx) 对象进行了实例化之后，将创建并初始化应用程序对象，如[HttpContext](https://msdn.microsoft.com/zh-cn/library/system.web.httpcontext(v=vs.100).aspx)、[HttpRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest(v=vs.100).aspx) 和 [HttpResponse](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse(v=vs.100).aspx)。[HttpContext](https://msdn.microsoft.com/zh-cn/library/system.web.httpcontext(v=vs.100).aspx) 类包含特定于当前应用程序请求的对象，如 [HttpRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest(v=vs.100).aspx) 和[HttpResponse](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse(v=vs.100).aspx) 对象。[HttpRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest(v=vs.100).aspx) 对象包含有关当前请求的信息，包括 Cookie 和浏览器信息。[HttpResponse](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse(v=vs.100).aspx) 对象包含发送到客户端的响应，其中包括所有呈现的输出和 Cookie。  下面是 IIS 6.0 和 IIS 7.0（在集成模式下，与 .NET Framework 3.0 或更高版本一起运行）之间的某些关键差异：  可以使用 [HttpResponse](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse(v=vs.100).aspx) 对象的 [SubStatusCode](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse.substatuscode(v=vs.100).aspx) 属性，设置对失败请求跟踪有用的代码。有关更多信息，请参见[Troubleshooting Failed Requests Using Failed Request Tracing in IIS 7.0](http://go.microsoft.com/fwlink/?LinkId=87359)（使用 IIS 7.0 中的失败请求跟踪功能解决失败请求存在的问题）。  [HttpResponse](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse(v=vs.100).aspx) 对象的 [Headers](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse.headers(v=vs.100).aspx) 属性提供对响应的响应头的访问  在一个事件处理程序处理多个 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 事件时，可以使用 [HttpContext](https://msdn.microsoft.com/zh-cn/library/system.web.httpcontext(v=vs.100).aspx) 对象的两个属性 [IsPostNotification](https://msdn.microsoft.com/zh-cn/library/system.web.httpcontext.ispostnotification(v=vs.100).aspx) 和[CurrentNotification](https://msdn.microsoft.com/zh-cn/library/system.web.httpcontext.currentnotification(v=vs.100).aspx)。  [HttpRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest(v=vs.100).aspx) 对象的 [Headers](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest.headers(v=vs.100).aspx) 和 [ServerVariables](https://msdn.microsoft.com/zh-cn/library/system.web.httprequest.servervariables(v=vs.100).aspx) 属性支持写操作。 |
| 将[HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 对象分配给请求 | 初始化所有应用程序对象之后，将通过创建 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 类的实例来启动应用程序。如果应用程序有 Global.asax 文件，则 ASP.NET 会创建从 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 类派生的 Global.aspx 类的实例。然后使用该派生类来表示应用程序。  说明**说明：**  第一次在应用程序中请求 ASP.NET 页或进程时，将创建 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 类的一个新实例。不过，为了尽可能提高性能，可对多个请求重复使用 [HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 实例。  加载哪些 ASP.NET 模块（如 [SessionStateModule](https://msdn.microsoft.com/zh-cn/library/system.web.sessionstate.sessionstatemodule(v=vs.100).aspx)）取决于应用程序从父应用程序继承的托管代码模块。这还取决于在应用程序的 Web.config 文件的配置节中配置了哪些模块。在应用程序的 Web.config 的 **system.webServer** 节中的 **modules** 元素中添加或移除模块。有关更多信息，请参见 [如何：为 IIS 7.0 配置 <system.webServer> 节](https://msdn.microsoft.com/zh-cn/library/bb763179(v=vs.100).aspx)。 |
| 由[HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 管线处理请求。 | 在处理请求时，[HttpApplication](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication(v=vs.100).aspx) 类会执行下列任务。这些事件对于希望在引发关键请求管道事件时运行代码的网页开发人员很有用。如果是在开发自定义模块，并且希望对发往管道的所有请求都调用该模块，则这些事件也很有用。自定义模块实现[IHttpModule](https://msdn.microsoft.com/zh-cn/library/system.web.ihttpmodule(v=vs.100).aspx) 接口。在 IIS 7.0 集成模式下，必须在模块的 [Init](https://msdn.microsoft.com/zh-cn/library/system.web.ihttpmodule.init(v=vs.100).aspx) 方法中注册事件处理程序。  对请求进行验证，将检查浏览器发送的信息，并确定其是否包含潜在恶意标记。有关更多信息，请参见 [ValidateRequest](https://msdn.microsoft.com/zh-cn/library/system.web.configuration.pagessection.validaterequest(v=vs.100).aspx)和[脚本侵入概述](https://msdn.microsoft.com/zh-cn/library/w1sw53ds(v=vs.100).aspx)。  如果已在 Web.config 文件的 [UrlMappingsSection](https://msdn.microsoft.com/zh-cn/library/system.web.configuration.urlmappingssection(v=vs.100).aspx) 节中配置了任何 URL，则执行 URL 映射。  引发 [BeginRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.beginrequest(v=vs.100).aspx) 事件  引发 [AuthenticateRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.authenticaterequest(v=vs.100).aspx) 事件。  引发 [PostAuthenticateRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postauthenticaterequest(v=vs.100).aspx) 事件。  引发 [AuthorizeRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.authorizerequest(v=vs.100).aspx) 事件。  引发 [PostAuthorizeRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postauthorizerequest(v=vs.100).aspx) 事  引发 [ResolveRequestCache](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.resolverequestcache(v=vs.100).aspx) 事件。  引发 [PostResolveRequestCache](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postresolverequestcache(v=vs.100).aspx) 事件。  引发 [MapRequestHandler](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.maprequesthandler(v=vs.100).aspx) 事件。将根据所请求资源的文件扩展名，选择相应的处理程序。处理程序可以是本机代码模块，如 IIS 7.0**StaticFileModule**，也可以是托管代码模块，如 [PageHandlerFactory](https://msdn.microsoft.com/zh-cn/library/system.web.ui.pagehandlerfactory(v=vs.100).aspx) 类（它处理 .aspx 文件）。  引发 [PostMapRequestHandler](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postmaprequesthandler(v=vs.100).aspx) 事件。  引发 [AcquireRequestState](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.acquirerequeststate(v=vs.100).aspx) 事件。  引发 [PostAcquireRequestState](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postacquirerequeststate(v=vs.100).aspx) 事件。  引发 [PreRequestHandlerExecute](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.prerequesthandlerexecute(v=vs.100).aspx) 事件。  为该请求调用合适的 [IHttpHandler](https://msdn.microsoft.com/zh-cn/library/system.web.ihttphandler(v=vs.100).aspx) 类的 [ProcessRequest](https://msdn.microsoft.com/zh-cn/library/system.web.ihttphandler.processrequest(v=vs.100).aspx) 方法（或异步版IHttpAsyncHandler.BeginProcessRequest）。例如，如果该请求针对某页，则当前的页实例将处理该请求。  引发 [PostRequestHandlerExecute](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postrequesthandlerexecute(v=vs.100).aspx) 事件。  引发 [ReleaseRequestState](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.releaserequeststate(v=vs.100).aspx) 事件。  引发 [PostReleaseRequestState](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postreleaserequeststate(v=vs.100).aspx) 事件。  如果定义了 [Filter](https://msdn.microsoft.com/zh-cn/library/system.web.httpresponse.filter(v=vs.100).aspx) 属性，则执行响应筛选。  引发 [UpdateRequestCache](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.updaterequestcache(v=vs.100).aspx) 事件。  引发 [PostUpdateRequestCache](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postupdaterequestcache(v=vs.100).aspx) 事件。  引发 [LogRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.logrequest(v=vs.100).aspx) 事件。  引发 [PostLogRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postlogrequest(v=vs.100).aspx) 事件。  引发 [EndRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.endrequest(v=vs.100).aspx) 事件。  引发 [PreSendRequestHeaders](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.presendrequestheaders(v=vs.100).aspx) 事件。  引发 [PreSendRequestContent](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.presendrequestcontent(v=vs.100).aspx) 事件。  说明**说明：**  只有当应用程序在 IIS 7.0 集成模式下运行，并且与 .NET Framework 3.0 或更高版本一起运行时，才会支持[MapRequestHandler](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.maprequesthandler(v=vs.100).aspx)、[LogRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.logrequest(v=vs.100).aspx) 和 [PostLogRequest](https://msdn.microsoft.com/zh-cn/library/system.web.httpapplication.postlogrequest(v=vs.100).aspx) 事件。 |