





**Worker Process**: **Worker Process** (w3wp.exe) runs the ASP.Net application in IIS. This **process** is responsible for managing all the request and response that are coming from the client system. All the ASP.Net functionality runs under the scope of the **worker process**.

Stands for "Internet Information Services." **IIS** is a web server software package designed for Windows Server. It is **used** for hosting websites and other content on the Web. Microsoft's Internet Information Services provides a graphical user interface (GUI) for managing websites and the associated users.11

Most commonly, **IIS** is used to host ASP.NET web applications and static websites. It can also be used as an FTP server, host WCF services, and be extended to host web applications built on other platforms such as PHP. There are built-in authentication options such as Basic, ASP.NET, and Windows auth.

**ASP.Net Page Life Cycle**

****

**Page request**

The page request occurs before the page life cycle begins. When the page is requested by a user, ASP.NET determines whether the page needs to be parsed and compiled (therefore beginning the life of a page), or whether a cached version of the page can be sent in response without running the page.

**Start**

In the start stage, page properties such as [Request](http://msdn.microsoft.com/en-us/library/system.web.ui.page.request(v=vs.100).aspx) and [Response](http://msdn.microsoft.com/en-us/library/system.web.ui.page.response(v=vs.100).aspx) are set. At this stage, the page also determines whether the request is a postback or a new request and sets the [IsPostBack](http://msdn.microsoft.com/en-us/library/system.web.ui.page.ispostback(v=vs.100).aspx) property. The page also sets the [UICulture](http://msdn.microsoft.com/en-us/library/system.web.ui.page.uiculture(v=vs.100).aspx) property.

**Initialization**

During page initialization, controls on the page are available and each control's [UniqueID](http://msdn.microsoft.com/en-us/library/system.web.ui.control.uniqueid(v=vs.100).aspx) property is set. A master page and themes are also applied to the page if applicable. If the current request is a postback, the postback data has not yet been loaded and control property values have not been restored to the values from view state.

**Load**

During load, if the current request is a postback, control properties are loaded with information recovered from view state and control state.

**Postback event handling**

If the request is a postback, control event handlers are called. After that, the [Validate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.basevalidator.validate(v=vs.100).aspx) method of all validator controls is called, which sets the [IsValid](http://msdn.microsoft.com/en-us/library/system.web.ui.ivalidator.isvalid(v=vs.100).aspx) property of individual validator controls and of the page. (There is an exception to this sequence: the handler for the event that caused validation is called after validation.)

**Rendering**

Before rendering, view state is saved for the page and all controls. During the rendering stage, the page calls the [Render](http://msdn.microsoft.com/en-us/library/system.web.ui.control.render(v=vs.100).aspx) method for each control, providing a text writer that writes its output to the [OutputStream](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.outputstream(v=vs.100).aspx) object of the page's [Response](http://msdn.microsoft.com/en-us/library/system.web.ui.page.response(v=vs.100).aspx) property.

**Unload**

The [Unload](http://msdn.microsoft.com/en-us/library/system.web.ui.control.unload(v=vs.100).aspx) event is raised after the page has been fully rendered, sent to the client, and is ready to be discarded. At this point, page properties such as [Response](http://msdn.microsoft.com/en-us/library/system.web.ui.page.response(v=vs.100).aspx) and [Request](http://msdn.microsoft.com/en-us/library/system.web.ui.page.request(v=vs.100).aspx) are unloaded and cleanup is performed.

**pre-init** : all controls created during the design time are initialized with the default values.  
**On-init**: Here we can read the control Properties where as you cannot read the control values.  
**Load view state**: This event gets triggers only if the page is post back (i.e. ispageback==true). loads all the controls which are viewstate enabled  
**LoadPostBackData**: This event also gets triggered only if the page is post-back.  
**PageLoad**: control properties are loaded with information recovered from view state and control state.  
**Control event handlers:** These events to handle specific control events,  
**PreRender**: This event is triggered recursively for each child control in the page  
**SaveViewState**: Raised after view state and control state have been saved for the page and for all controls  
**Render** : This method writes out the control’s markup to send to the browser.  
**unload** : This event does the final cleanup for controls, used in the page

For more details:

http://msdn.microsoft.com/en-us/library/ms178472(v=vs.100).aspx