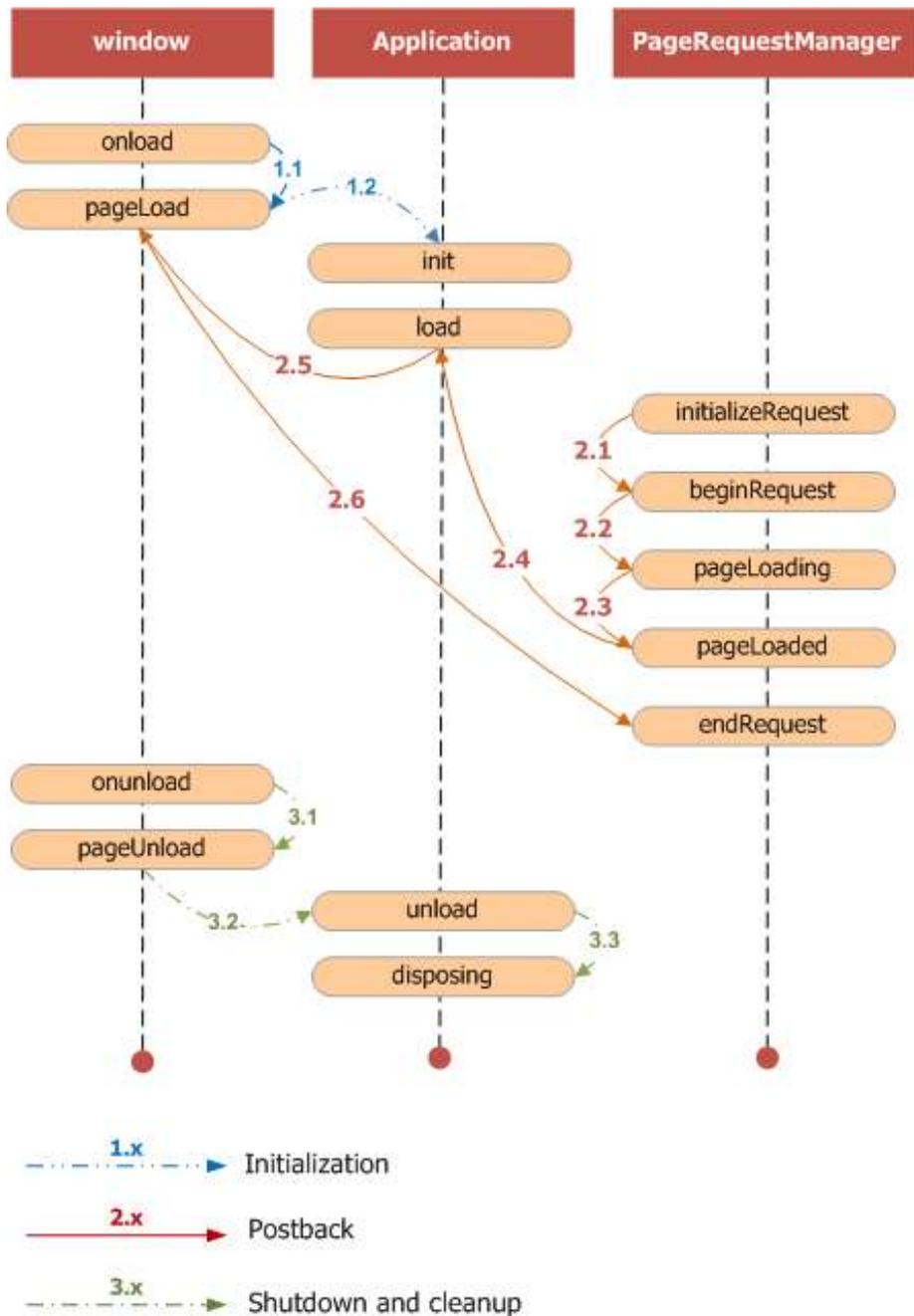


Microsoft ASP.NET AJAX Client Life-Cycle Events



The two main Microsoft AJAX Library classes that raise events during the client life cycle of a page are the **Application** and **PageRequestManager** classes.

The key event for initial requests (GET requests) and synchronous postbacks is the **load** event of the **Application** instance. When script in a **load** event handler runs, all scripts and components have been loaded and are available.

When partial-page rendering with **UpdatePanel** controls is enabled, the key client events are the events of the **PageRequestManager** class. These events enable you to handle many common scenarios. These include the ability to cancel postbacks, to give precedence to one postback over another, and to animate **UpdatePanel** controls when their content is refreshed.

To add or remove handlers for events raised by the **Application** and **PageRequestManager** classes, use the **add_eventname** and **remove_eventname** methods of those classes.

Note: To handle the **load** and **unload** events of the **Application** object, you can create functions that use the reserved names **pageLoad** and **pageUnload**.

```
function pageLoad() {
    var prm = Sys.WebForms.PageRequestManager.getInstance();

    if (!prm.get_isInAsyncPostBack())
        prm.add_beginRequest (onBeginRequest);
}
```

Sys.WebForms.PageRequestManager Events

initializeRequest	Raised during initialization of an asynchronous postback.
beginRequest	Raised before processing of an asynchronous postback starts, and the postback request is sent to the server. If there is a postback already processing, it is stopped using the abortPostBack method.
pageLoading	Raised after a response to an asynchronous postback is received from the server, but before any content on the page is updated.
pageLoaded	Raised after all content on the page is refreshed as the result of either a synchronous or an asynchronous postback.
endRequest	Raised after an asynchronous postback is finished and control has been returned to the browser. If an error occurs, the page is not updated. Use this event to provide customized error notification to users or to log errors.

Sys.Application Events

init	Raised only once when the page is first rendered after all scripts have been loaded, but before objects are created.
load	Raised after all scripts have been loaded and objects in the application have been created and initialized. Raised also for all postbacks to the server, which includes asynchronous postbacks.
unload	Raised before all objects in the client application are disposed. During this event you should free any resources that your code is holding.