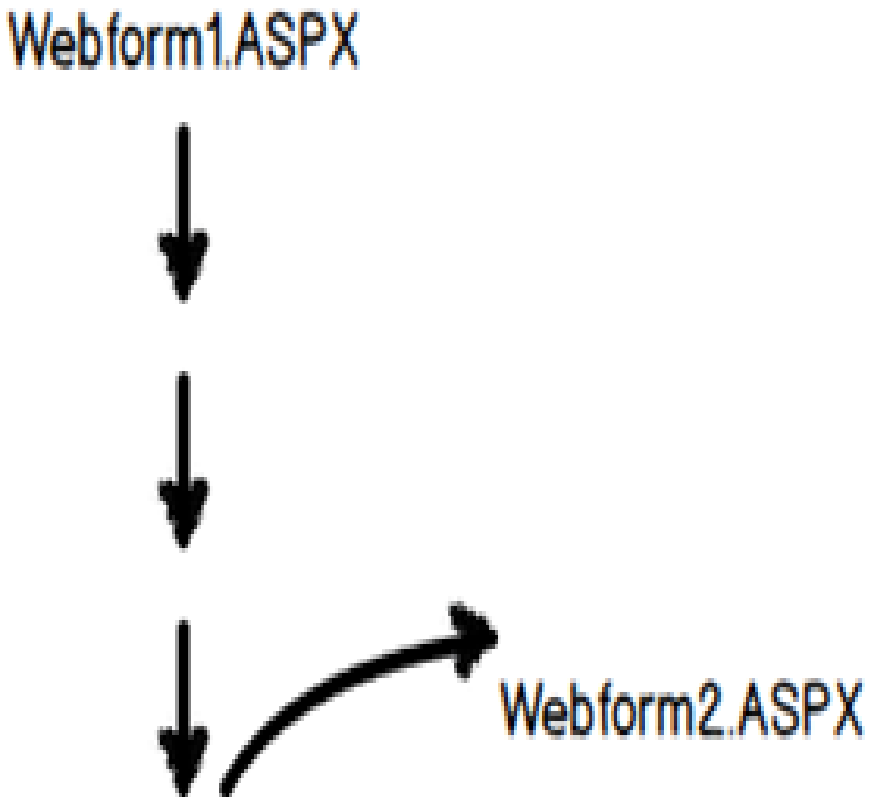
Server.Transfer VS Response.Redirect – Simplified

Introduction

In ASP.NET some of the concepts do the same task but are meant to be used in different scenarios. One such concept which is confusing and most discussed among developers is the difference between “Server.Transfer” and “Response.Redirect”.



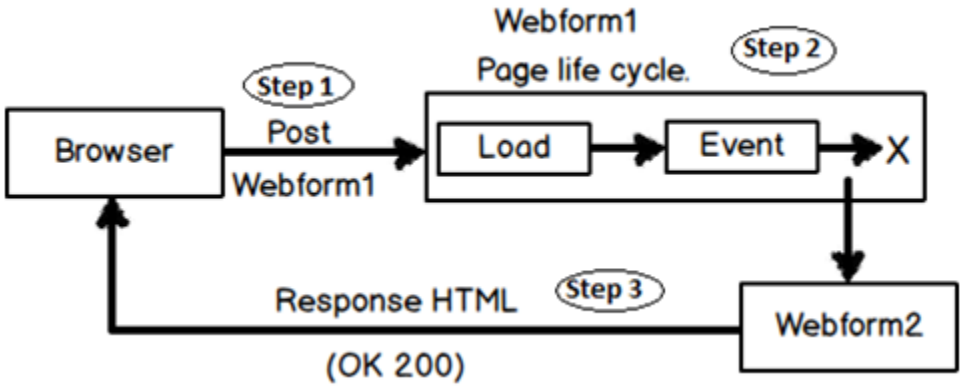
“response.redirect” and “server.transfer” helps to transfer user from one page to other page while the page is executing. But the way they do this transfer / redirect is very different.

“Server.Transfer”vs “response.Redirect”

The main difference between them is who does the transfer. In “response.redirect” the transfer is done by the browser while in “server.transfer” it’s done by the server. Let us try to understand this statement in a more detail manner.

In “Server.Transfer” following is the sequence of how transfer happens:-

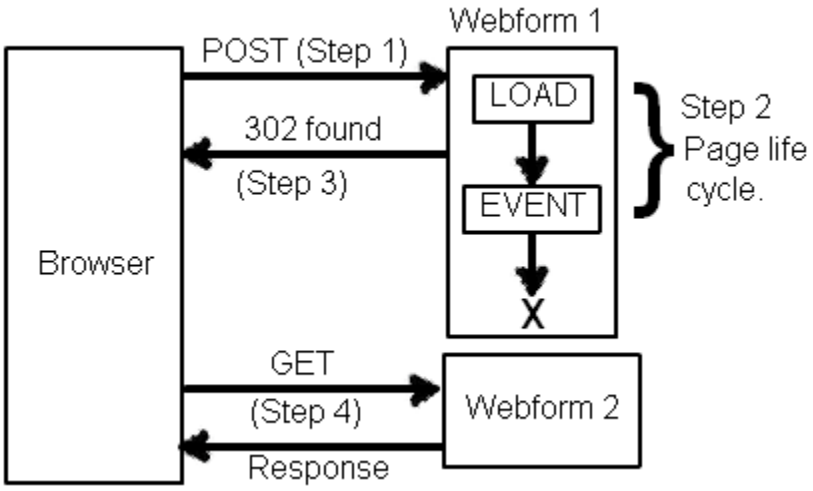
1. User sends a request to an ASP.NET page. In the below figure the request is sent to “WebForm1” and we would like to navigate to “Webform2”.
2. Server starts executing “Webform1” and the life cycle of the page starts. But before the complete life cycle of the page is completed “Server.transfer” happens to “WebForm2”.
3. “Webform2” page object is created, full page life cycle is executed and output HTML response is then sent to the browser.



One important point to note here is the URL is not changed to the target page. If you have sent request from “Webform1.aspx” to redirect to “WebForm2.aspx” on the browser URL you will still see “WebForm1.aspx”.

While in “Response.Redirect” following is the sequence of events for navigation:-

1. Client (browser) sends a request to a page. In the below figure the request is sent to “WebForm1” and we would like to navigate to “Webform2”.
2. Life cycle of “Webform1” starts executing. But in between of the life cycle “Response.Redirect” happens.
3. Now rather than server doing a redirect , he sends a HTTP 302 command to the browser. This command tells the browser that he has to initiate a GET request to “Webform2.aspx” page.
4. Browser interprets the 302 command and sends a GET request for “Webform2.aspx”.

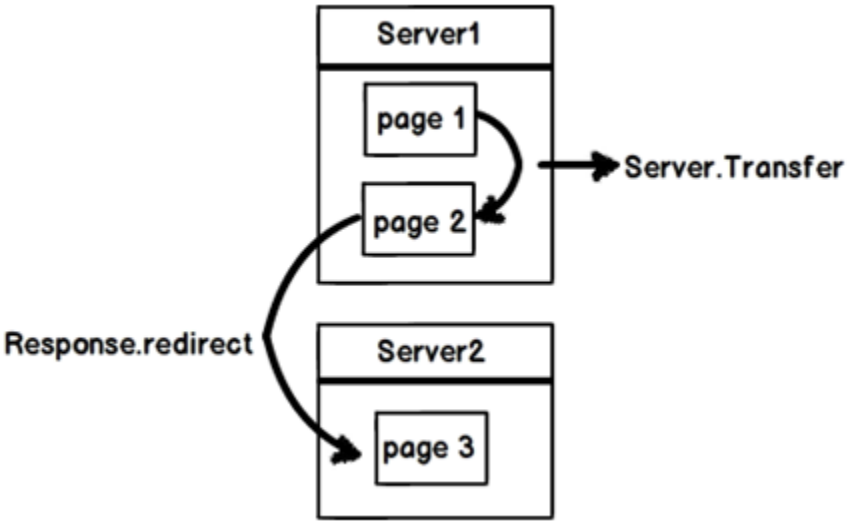


In this case you will the URL's are changed as per redirection. So if you have redirected to "Webform2.aspx" then on the browser URL you should see "WebForm2.aspx".

In other words "Server.Transfer" is executed by the server while "Response.Redirect" is executed by thr browser. "Response.Redirect" needs to two requests to do a redirect of the page.

So when to use "Server.Transfer" and when to use "Response.Redirect" ?

Use "Server.Transfer" when you want to navigate pages which reside on the same server, use "Response.Redirect" when you want to navigate between pages which resides on different server and domain.



Below goes the consolidated table with all the differences as discussed at the top.

	Server.Transfer	**Response.Redirect**
Redirection	Redirection is done by the server.	Redirection is done by the browser client.
Browser URL	Does not change.	Changes to the redirected target page.
When to use	Redirect between pages of the same server.	Redirect between pages on different server and domain.

What is importance of "preserveForm" flag in "Server.Transfer"?

"Server.Transfer" helps to redirect from one page to other page. If you wish to pass query string and form data of the first page to the target page during this redirection you need to set "preserveForm" to "true" as shown in the below code.

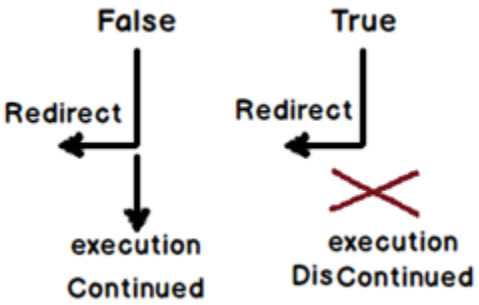
```
Server.Transfer("Webform2.aspx",true);
```

By default the value of "preserveForm" is "true".

Response.Redirect(URL,true) vsResponse.Redirect(URL,false) ?

Response.Redirect(URL,false) :- Client is redirected to a new page and the current page on the server will keep processing ahead.

Response.Redirect(URL,true) :- Client is redirected to a new page but the processing of the current page is aborted.



Below is a facebook video whichdemonstrates practically the difference between server.transfervsresponse.redirect .

Written by Shivprasad koirala