

Here are a few tips to improve the performance of your ASP.Net application.

1. **Viewstate**

View state is the wonder mechanism that shows the details of the entry posted on the server. It is loaded every time ifrom the server. This option looks like an extra feature for the end users. This needs to be loaded from the server and it adds more size to the page but it will affect the performance when we have many controls in the page, like user registration. So, if there is no need for it then it can be disabled.

EnableViewState = "false" needs to be given based on the requirements. It can be given at the control, page and config level settings.

2. **Avoid Session and Application Variables**

A Session is a storage mechanism that helps developers to take values across the pages. It will be stored based on the session state chosen. By default it will be stored in the Inproc. That default settings uses IIS. When this Session variable is used in a page that is accessed by many numbers then it will occupy more memory allocation and gives additional overhead to the IIS. It will make the performance slow.

It can be avoided for most scenarios . If you want to send the information across pages then we can use a Cross Post-back, Query string with encryption. If you want to store the information within the page then caching the object is the best way.

3. **Use Caching**

ASP.Net has the very significant feature of a caching mechanism. It gives more performance and avoids the client/server process. There are three types of caching in ASP.Net.

If there is any static content in the full pages then it should be used with the Output cache. What it does is, it stores the content on IIS. When the page is requested it will be loaded immediately from the IIS for the certain period of time. Similarly Fragment paging can be used to store the part of the web page.

4. **Effectively use CSS and Script files**

If you have large CSS files that are used for the entire site in multiple pages, then based on the requirements, it can be split and stored with different names. It will minimize the loading time of the pages.

5. **Images sizes**

Overuse of images in the web site affect the web page performance. It takes time to load the images, especially on dial-up connections. Instead of using the background images, it can be done on the CSS colors or use light-weight images to be repeated in all of the pages.

6. **CSS based layout**

The entire web page design is controlled by the CSS using the div tags instead of table layout. It increases the page loading performance dramatically. It will help to enforce the same standard guideline throughout the website. It will reduce the future changes easily. When we use the nested table layout it takes more time for rendering.

7. **Avoid Round trips**

We can avoid unnecessary database hits to load the unchanged content in the database. We should use the IsPostBack method to avoid round trips to the database.

8. **Validate using JavaScript**

Manual validation can be done at the client browser instead of doing at the server side. JavaScript helps us to do the validation at the client side. This will reduce the additional overhead to the server.

The plug-in software helps to disable the coding in the client browser. So, the sensitive application should do the server side validation before going into the process.

9. **Clear the Garbage Collection**

Normally .Net applications use Garbage Collection to clean the unused resources from memory. But it takes time to clear the unused objects from memory.

There are many ways to clean the unused resources. But not all of the methods are recommended. But we can use the dispose method in the finally block to clean up the resources. Moreover we need to close the connection. It will immediately free the resources and provides space in memory.

10. **Avoid bulk data store on client side**

Try to avoid more data on the client side. It will affect the web page loading. When we store more data on the hidden control then it will be encrypted and stored on the client side. It can be tampered with by hackers as well.

11. **Implement Dynamic Paging**

When we load a large number of records into the server data controls like GridView, DataList and ListView it will take time to load. So we can show only the current page data through the dynamic paging.

12. **Use Stored Procedure**

Try to use Stored Procedures. They will increase the performance of the web pages. Because it is stored as a compiled object in the database and it uses the query execution plans. If you pass the query then it will make a network query. In the Stored Procedure a single line will be passed to

the backend.

13. **Use XML and XSLT**

XML and XSLT will speed up the page performance. If the process is not more complex then it can be implemented in XSLT.

14. **Use Dataset**

A DataSet is not lightweight compared with DataReader. But it has the advantages of a disconnected architecture. A DataSet will consume a substantial amount of memory. Even though it can have more than one day. If you want to perform many operations while loading the page itself, then it might be better to go with a DataSet. Once data is loaded into the DataSet it can be used later also.

15. **Use String Builder in place of String**

When we append the strings like mail formatting in the server side then we can use StringBuilder. If you use string for concatenation, what it does every time is it creates the new storage location for storing that string. It occupies more spaces in memory. But if we use the StringBuilder class in C# then it consumes more memory space than String.

16. **Use Server.Transfer**

If you want to transfer the page within the current server then we can use the Server.Transfer method. It avoids roundtrips between the browser and server. But it won't update the browser history.

17. **Use Threads**

Threads are an important mechanism in programming to utilize the system resources effectively. When we want to do a background process then it can be called a background process.

Consider the example of when clicked on send, it should send the mail to 5 lakhs members yet there is no need to wait for all the processes to complete. Just call the mail sending process as a background thread then proceed to do the further processing, because the sending of the mail is not dependent on any of the other processes.