

# Weird session state problem, maybe?

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2007-04/msg00586.html>

---

- *From:* vincentw56 <[MSDNNospam252@xxxxxxxxxxxxxx](mailto:MSDNNospam252@xxxxxxxxxxxxxx)>
  - *Date:* Wed, 25 Apr 2007 06:52:01 -0700
- 

I have an ASP.NET 2.0 application. It is pretty basic. What it does is shows a gridview of data from a stored procedure. The user can also select a date filter. The problem is that sometimes an error is thrown about getting the data. If I go in and make a change to the .cs file, the application runs fine. The change I make is like insert a space somewhere just to make the file recompile and it works again. What causes this? It sounds like the session state is persisting or something. I have posted the code below. Any help would be appreciated. Thanks.

```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Security.Principal;
using System.Text;
using NsiMsCrmWebService;
using System.Threading;
using System.Collections;

public partial class _Default : System.Web.UI.Page
{
    public static DataSet myDataSet = new DataSet();
    double dblGroupSumOppTotal = 0, dblGroupSumExpected = 0,
    dblGroupSumForecast = 0, dblReportSum = 0, dblGroupSumDirForecast = 0;
    NsiMsCrmWebService.NsiMsCrmWebService myWebService = new
    NsiMsCrmWebService.NsiMsCrmWebService();
    static string strTerritory = "", strStartDate = "", strEndDate = "",
    strSort = "", strSortOrder = "";

    #region Page_Load
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
```

## Weird session state problem, maybe?

```
{
strStartDate = "";
strEndDate = "";

pnlReport.Visible = false;
pnlSelectTerritory.Visible = false;
pnlSearch.Visible = false;
lbnFilter.Visible = false;

//First we need to see if they are authorized to use this tool
WindowsPrincipal p = Thread.CurrentPrincipal as WindowsPrincipal;
int idx = p.Identity.Name.IndexOf('\\');
string strUserName = p.Identity.Name.Substring(idx + 1);

//Now that we have the username, we need to check to see if they
are authorized
//and display the right territory
switch (strUserName.ToLower())
{
//Eastern Rep
case "chris.montanaro":
//Build the list item
ddlTerritory.Items.Add(new ListItem("Canada", "Canada"));
ddlTerritory.Items.Add(new ListItem("Eastern",
"Eastern"));
pnlSelectTerritory.Visible = true;
break;

//Western Rep
case "steve.dupree":
//Build the list item
ddlTerritory.Items.Add(new ListItem("Canada", "Canada"));
ddlTerritory.Items.Add(new ListItem("Western",
"Western"));
pnlSelectTerritory.Visible = true;
break;

//Admins
case "dean.goodermote":
case "william.waller":
case "bob.davis":
case "will.anderson":
case "dan.jones":
case "donna.colonna":
case "vwright":
case "vincent.wright":
case "anna.combs":
case "melinda.denney":
ddlTerritory.Items.Add(new ListItem("Canada", "Canada"));
ddlTerritory.Items.Add(new ListItem("Eastern",
"Eastern"));
```

Weird session state problem, maybe?

## Weird session state problem, maybe?

```
ddlTerritory.Items.Add(new ListItem("Western",
"Western"));
ddlTerritory.Items.Add(new ListItem("International",
"International"));
pnlSelectTerritory.Visible = true;
break;

//No Access
default:
lblStatus.Text = "You do not have access to this
application.";
break;
}
}
}
#endregion

#region BindData()
void BindData()
{
if (strTerritory != "")
{
myDataSet = myWebService.rdfGetData(strTerritory, strStartDate,
strEndDate);

myDataSet.Tables[0].TableName = "Data";
myDataSet.Tables.Add("ESM");

string[] distinct = { "ESM" };

myDataSet.Tables["ESM"].Clear();

myDataSet.Tables["ESM"].Merge(myDataSet.Tables["Data"].DefaultView.ToTable(true, distinct));
myDataSet.Relations.Add(new DataRelation("ESMGroup",
myDataSet.Tables["ESM"].Columns["ESM"],
myDataSet.Tables["Data"].Columns["ESM"]));

myRepeater.DataSource = myDataSet.Tables["ESM"];
myRepeater.DataBind();

lblnFilter.Visible = true;
}
}
#endregion

#region myRepeater_ItemDataBound
protected void myRepeater_ItemDataBound(object sender,
RepeaterItemEventArgs e)
{
if (e.Item.ItemType == ListItemType.Item || e.Item.ItemType ==
ListItemType.AlternatingItem)
```

Weird session state problem, maybe?

## Weird session state problem, maybe?

```
{
Repeater rpt = (Repeater)e.Item.FindControl("myRepeater2");
if (rpt != null)
{
DataView myDataView =
((DataRowView)e.Item.DataItem).CreateChildView("ESMGroup");

if (! string.IsNullOrEmpty(strSort))
{
myDataView.Sort = strSort + " " + strSortOrder;
}

rpt.DataSource = myDataView;
rpt.DataBind();
}
}
}
#endregion

#region myRepeater2_ItemDataBound
protected void myRepeater2_ItemDataBound(object sender,
RepeaterItemEventArgs e)
{
if (e.Item.ItemType == ListItemType.Item || e.Item.ItemType ==
ListItemType.AlternatingItem)
{
string strEstimatedValue = DataBinder.Eval(e.Item.DataItem,
"estimatedValue").ToString();
bool bEsmForecast = (bool)DataBinder.Eval(e.Item.DataItem,
"CFBForecast");
bool bDirForecast = (bool)DataBinder.Eval(e.Item.DataItem,
"CFBDirectorForecast");

if (strEstimatedValue != "")
{
dblGroupSumExpected +=
Convert.ToDouble(DataBinder.Eval(e.Item.DataItem, "estimatedValue"));
}

if (bEsmForecast)
{
dblGroupSumForecast +=
Convert.ToDouble(DataBinder.Eval(e.Item.DataItem, "estimatedValue"));
}

if (bDirForecast)
{
CheckBox myCheckBox =
(CheckBox)e.Item.FindControl("chkForecast");
myCheckBox.Checked = true;
dblGroupSumDirForecast +=
```

Weird session state problem, maybe?

## Weird session state problem, maybe?

```
Convert.ToDouble(DataBinder.Eval(e.Item.DataItem, "estimatedValue"));
}
```

```
dblGroupSumOppTotal += 1;
```

```
//We need to set the image for the ESM forecast
if (bEsmForecast)
{
    HtmlTableCell myTableCell =
    (HtmlTableCell)e.Item.FindControl("tdEsmForecast");
    myTableCell.InnerHtml = "<strong>X</strong>";
}
}
else
{
    if (e.Item.ItemType == ListItemType.Footer)
    {
        //Display the Total Expected Revenue (ESM) in the group footer
        Label groupSumExpected =
        (Label)e.Item.FindControl("lblGroupSumExpected");
        groupSumExpected.Text = dblGroupSumExpected.ToString("c");
        dblGroupSumExpected = 0;
```

```
//Display the Total Forecasted Revenue (ESM) in the group
footer
Label groupGroupSumForecast =
(Label)e.Item.FindControl("lblGroupSumForecast");
groupGroupSumForecast.Text =
dblGroupSumForecast.ToString("c");
dblGroupSumForecast = 0;
```

```
//Display the Total Opportunities in the group footer
Label groupSumOppTotal =
(Label)e.Item.FindControl("lblGroupSumOppTotal");
groupSumOppTotal.Text = dblGroupSumOppTotal.ToString();
dblGroupSumOppTotal = 0;
```

```
//Display the Total Opportunities in the group footer
Label groupGroupSumDirForecast =
(Label)e.Item.FindControl("lblGroupSumDirForecast");
groupGroupSumDirForecast.Text =
dblGroupSumDirForecast.ToString("c");
dblReportSum += dblGroupSumDirForecast;
dblGroupSumDirForecast = 0;
}
}
```

```
lblReportSum.Text = strTerritory + " - " + dblReportSum.ToString("c");
lblForecastTotal.Text = dblReportSum.ToString("c");
}
#endregion
```

Weird session state problem, maybe?

## Weird session state problem, maybe?

```
#region btnUpdate_Click
protected void btnUpdate_Click(object sender, EventArgs e)
{

    DataSet dsUpdateForecast = new DataSet();
    DataTable dtUpdateForecast = new DataTable();
    dtUpdateForecast.Columns.Add("opportunityid");
    dtUpdateForecast.Columns.Add("directorsforecast");

    myDataSet.Tables["Data"].DefaultView.Sort = "opportunityid,
    CFBDirectorForecast";

    for (int i = 0; i < myRepeater.Items.Count; i++)
    {
        Repeater detailRepeater =
        (Repeater)myRepeater.Items[i].FindControl("myRepeater2");

        for (int j = 0; j < detailRepeater.Items.Count; j++)
        {
            CheckBox directorForecast =
            (CheckBox)detailRepeater.Items[j].FindControl("chkForecast");
            Literal rdfGuidLiteral =
            (Literal)detailRepeater.Items[j].FindControl("rdfGuid");

            object[] objVal = new object[2];
            objVal[0] = rdfGuidLiteral.Text;
            objVal[1] = directorForecast.Checked.ToString();

            int iFound =
            myDataSet.Tables["Data"].DefaultView.Find(objVal);

            if (iFound < 0)
            {
                DataRow myDataRow = dtUpdateForecast.NewRow();
                myDataRow["opportunityid"] = rdfGuidLiteral.Text;
                myDataRow["directorsforecast"] =
                directorForecast.Checked.ToString();
                dtUpdateForecast.Rows.Add(myDataRow);
            }
        }
    }

    dsUpdateForecast.Tables.Add(dtUpdateForecast);

    myWebService.Timeout = 600000;
    bool bUpdate = myWebService.rdfSetForecast(dsUpdateForecast);

    if (bUpdate)
    {
        lblStatus.Text = "The Report was successfully updated";
    }
}

```

Weird session state problem, maybe?

Weird session state problem, maybe?

```
}
else
{
lblStatus.Text = "There was an error updating the report.";
}

BindData();

}
#endregion

#region btnSubmitTerritory_Click
protected void btnSubmitTerritory_Click(object sender, EventArgs e)
{
strTerritory = ddlTerritory.SelectedValue;
pnlSelectTerritory.Visible = false;
pnlReport.Visible = true;
strStartDate = "";
strEndDate = "";
BindData();
}
#endregion

#region lblFilter_Click
protected void lblFilter_Click(object sender, EventArgs e)
{
pnlReport.Visible = false;
pnlSearch.Visible = true;
lblStatus.Text = "";
}
#endregion

#region btnSearchSubmit_Click
protected void btnSearchSubmit_Click(object sender, EventArgs e)
{
strStartDate = txtStartDate.Text;
strEndDate = txtEndDate.Text;

pnlReport.Visible = true;
pnlSearch.Visible = false;

BindData();
}
#endregion

#region btnSearchCancel_Click
protected void btnSearchCancel_Click(object sender, EventArgs e)
{
pnlReport.Visible = true;
BindData();
}
#endregion
```

Weird session state problem, maybe?

## Weird session state problem, maybe?

```
#endregion

#region Sort_Click
protected void Sort_Click(object sender, EventArgs e)
{
    LinkButton myLinkButton = (LinkButton)sender;

    switch (myLinkButton.ID)
    {
        case "OpportunitySort":
            strSort = "accountname";
            break;

        case "ExpectedRevenueSort":
            strSort = "estimatedValue";
            break;

        case "ExpectedCloseDateSort":
            strSort = "estimatedClosedate";
            break;

        case "EsmForecastSort":
            strSort = "CFBForecast";
            break;

        case "DirectorsForecastSort":
            strSort = "CFBDirectorForecast";
            break;

        default:
            break;
    }

    if (strSortOrder == "ASC")
    {
        strSortOrder = "DESC";
    }
    else
    {
        strSortOrder = "ASC";
    }

    BindData();
}
#endregion
--
Vincent Wright
.
```