**Sample Session-State Store Provider**

This page is specific to

**Microsoft Visual Studio 2008/.NET Framework 3.5**

Other versions are also available for the following:

[Microsoft Visual Studio 2005/.NET Framework 2.0](http://msdn.microsoft.com/en-us/library/ms178588(VS.80).aspx)

[.NET Framework 3.0](http://msdn.microsoft.com/en-us/library/ms178588(VS.85).aspx)

[Microsoft Visual Studio 2010/.NET Framework 4](http://msdn.microsoft.com/en-us/library/ms178588(VS.100).aspx)

Describes a custom session-state store-provider implementation that uses the ODBC .NET Framework data provider to manage session information in an Access database.

The following topics include the code for a sample session-state store-provider implementation. The sample provider uses the [System.Data.Odbc](http://msdn.microsoft.com/en-us/library/system.data.odbc.aspx) classes to store and retrieve session information by using an Access database.

This topic describes implementation details about the sample session-state store provider and describes how to build the sample and configure an ASP.NET application to use the sample provider.

The code for the sample provider can be found in the [How to: Sample Session-State Store Provider](http://msdn.microsoft.com/en-us/library/ms178589.aspx) topic.

http://i.msdn.microsoft.com/Global/Images/clear.gif Database Schema

The sample session-state provider uses a single table named Sessions to manage session information. To create the Access table used by the sample provider, issue the following data-definition query in a new or existing Access database.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl25_ctl00_ctl00_code');" \o "Copy Code)

CREATE TABLE Sessions

(

SessionId Text(80) NOT NULL,

ApplicationName Text(255) NOT NULL,

Created DateTime NOT NULL,

Expires DateTime NOT NULL,

LockDate DateTime NOT NULL,

LockId Integer NOT NULL,

Timeout Integer NOT NULL,

Locked YesNo NOT NULL,

SessionItems Memo,

Flags Integer NOT NULL,

CONSTRAINT PKSessions PRIMARY KEY (SessionId, ApplicationName)

)

http://i.msdn.microsoft.com/Global/Images/clear.gif Event Log Access

If the sample provider encounters an exception when working with the data source, it writes the details of the exception to the Application Event Log instead of returning the exception to the ASP.NET application. This is done as a security measure to avoid private information about the data source from being exposed in the ASP.NET application.

The sample provider specifies an event [Source](http://msdn.microsoft.com/en-us/library/system.diagnostics.eventlog.source.aspx) property value of "OdbcSessionStateStore." Before your ASP.NET application will be able to write to the Application Event Log successfully, you will need to create the following registry key:

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl26_ctl00_ctl01_code');" \o "Copy Code)

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Eventlog\Application\OdbcSessionStateStore

If you do not want the sample provider to write exceptions to the event log, then you can set the custom **writeExceptionsToEventLog** attribute to **false** in the Web.config file.

http://i.msdn.microsoft.com/Global/Images/clear.gif Support for the Session\_OnEnd Event

The sample session-state store provider does not support the **Session\_OnEnd** event defined in the Global.asax file, as there is no way for the Access database to notify the session-state store provider that the expiration date and time for a session has passed. The session-state store provider must query for this information. You cannot predict when the session-state store provider will be used, and it is therefore unlikely that the **Session\_OnEnd** event will be raised at the exact time that the session times out. As a result, the [SetItemExpireCallback](http://msdn.microsoft.com/en-us/library/system.web.sessionstate.sessionstatestoreproviderbase.setitemexpirecallback.aspx) method implementation in the sample session-state store provider returns **false**, to inform the [SessionStateModule](http://msdn.microsoft.com/en-us/library/system.web.sessionstate.sessionstatemodule.aspx) that the **Session\_OnEnd** event is not supported.

http://i.msdn.microsoft.com/Global/Images/clear.gif Cleaning Up Expired Session Data

Because the sample session-state store provider does not provide support for the **Session\_OnEnd** event, it does not automatically clean up expired session-item data. It is recommended that you periodically delete expired session information from the data store with the following code.

Visual Basic

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl28_ctl00_ctl00_code');" \o "Copy Code)

Dim commandString As String = "DELETE FROM Sessions WHERE Expires < ?"

Dim conn As OdbcConnection = new OdbcConnection(connectionString)

Dim cmd As OdbcCommand = New OdbcCommand(commandString, conn)

cmd.Parameters.Add("@Expires", OdbcType.DateTime).Value = DateTime.Now

conn.Open()

cmd.ExecuteNonQuery()

conn.Close()

C#

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl28_ctl00_ctl01_code');" \o "Copy Code)

string commandString = "DELETE FROM Sessions WHERE Expires < ?";

OdbcConnection conn = new OdbcConnection(connectionString);

OdbcCommand cmd = new OdbcCommand(commandString, conn);

cmd.Parameters.Add("@Expires", OdbcType.DateTime).Value = DateTime.Now;

conn.Open();

cmd.ExecuteNonQuery();

conn.Close();

http://i.msdn.microsoft.com/Global/Images/clear.gif Building the Sample Provider

In order to use the sample provider, you can place your source code in the App\_Code directory of your application. Note that if you already have source code in the App\_Code directory of your application, you must add the version of the sample provider that is written in the same language as the existing code in the directory. The provider will be compiled by ASP.NET when your application is requested.

You can also compile the sample provider as a library and place it in the Bin directory of your Web application, or strongly name it and place it in the GAC. The following command shows how to compile the sample provider using the command-line compiler after the sample code has been copied to a file named OdbcSessionStateStore.vb for Visual Basic and OdbcSessionStateStore.cs for C#.

Visual Basic

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl29_ctl00_ctl00_code');" \o "Copy Code)

vbc /out:OdbcSessionStateStore.dll /t:library OdbcSessionStateStore.vb /r:System.Web.dll /r:System.Configuration.dll

C#

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl29_ctl00_ctl01_code');" \o "Copy Code)

csc /out:OdbcSessionStateStore.dll /t:library OdbcSessionStateStore.cs /r:System.Web.dll /r:System.Configuration.dll

http://i.msdn.microsoft.com/Global/Images/clear.gif Using the Sample Provider in an ASP.NET Application

The following example shows the Web.config file for an ASP.NET application configured to use the sample provider. The example uses an ODBC DSN named "SessionState" to obtain connection information for the Access database. To use the sample provider, you will need to either create the "SessionState" System DSN or supply a valid ODBC connection string to your database.

The example configuration assumes that your Web site is set up to use forms authentication and includes an ASP.NET page called login.aspx that allows users to log in.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl30_ctl00_ctl00_code');" \o "Copy Code)

<configuration>

<connectionStrings>

<add name="OdbcSessionServices" connectionString="DSN=SessionState;" />

</connectionStrings>

<system.web>

<sessionState

cookieless="true"

regenerateExpiredSessionId="true"

mode="Custom"

customProvider="OdbcSessionProvider">

<providers>

<add name="OdbcSessionProvider"

type="Samples.AspNet.Session.OdbcSessionStateStore"

connectionStringName="OdbcSessionServices"

writeExceptionsToEventLog="false" />

</providers>

</sessionState>

</system.web>

</configuration>