Configuration Sources Hands-On Lab for Enterprise Library

This walkthrough should act as your guide for learning how to configure Enterprise Library applications and how to leverage concepts such as configuration sources.

This lab comes with a sample application, the puzzler. When the sample application is run, two instances of the same program will be opened, each using different configuration files. The title of the application indicates the file from which its configuration originated.

The puzzler sample application uses both the Logging Application Block and the Exception Handling Application Block. Please refer to the Hands-On Labs for these application blocks in order to familiarize yourself with each and the concepts they are based on.

After completing this lab, you will be able to do the following:

- Share Enterprise Library between multiple application instances.
- Share Enterprise Library configuration sections between multiple application instances.
- Reuse an Enterprise Library configuration by means of inheritance.

This lab includes the following three exercises:

- Lab 1: Storing Enterprise Library configuration in a shared location.
- Lab 2: Redirecting an Enterprise Library configuration section to a shared location.
- Lab 3: Inheriting Enterprise Library configuration from a shared location.

The estimated completion for this lab is **15 minutes**.

Authors

These Hands-On Labs were produced by the following individuals:

- Product/Program Management: Grigori Melnik (Microsoft Corporation)
- Development: Chris Tavares (Microsoft Corporation), Nicolas Botto (Digit Factory), Olaf Conijn (Olaf Conijn BV), Fernando Simonazzi (Clarius Consulting), Erik Renaud (nVentive Inc.)
- Testing: Rick Carr (DCB Software Testing, Inc) plus everyone above
- Documentation: Alex Homer and RoAnn Corbisier (Microsoft Corporation) and Dennis DeWitt (Linda Werner & Associates Inc)

Lab 1: Storing Enterprise Library configuration in a shared location

This lab demonstrates how to store Enterprise Library configuration in a shared location by using the Enterprise Library configuration tools.
To begin this exercise, open the Puzzler2.sln file located in the ex01\begin folder.

To review the application

In the Microsoft® Visual Studio® solution for the project **PuzzlerUI**, notice that the project contains two configuration files, Puzzler1.config and Puzzler2.config. Both configuration files contain identical Enterprise Library configuration sections for the Exception Handling Application Block and Logging Application Block.

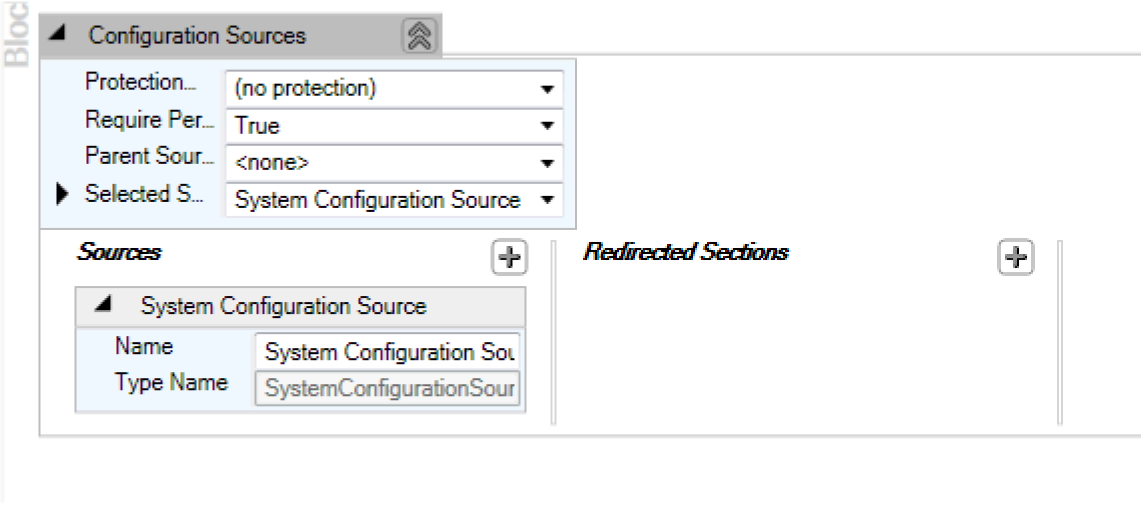
If you run the application in Visual Studio (**Debug** | **Start Debugging**), two application windows will open. Each application window will specify its respective configuration file in the title bar.

In both application windows, if you enter a word in the **Word To check** field and press the **Add Word** button, the application will display a message box containing information about a **SecurityException**. This exception is logged twice in the Windows® Event Log, once with the original exception message and once with a modified exception message, as specified in the Exception Handling Application Block's configuration. Open Event Viewer and inspect logs under Windows Logs | Application from the "Enterprise Library Logging" source.

To save Enterprise Library configuration to a shared file

1. In the PuzzlerUI project, right-click the Puzzler1.config file and click **Edit Enterprise Library V5 Configuration**. This will open the Enterprise Library V5 configuration editor inside Visual Studio.

2. On the Enterprise Library configuration editor **Blocks** menu select **Add Configuration Settings**. This will add the **Configuration Sources** section to the configuration file. You can expand the section's settings by clicking on the chevron icon to the right of the text **Configuration Sources**.



Configuration Sources

Protection...	(no protection)
Require Per...	True
Parent Sour...	<none>
Selected S...	System Configuration Source

Sources

System Configuration Source

Name	System Configuration Sou
Type Name	SystemConfigurationSour

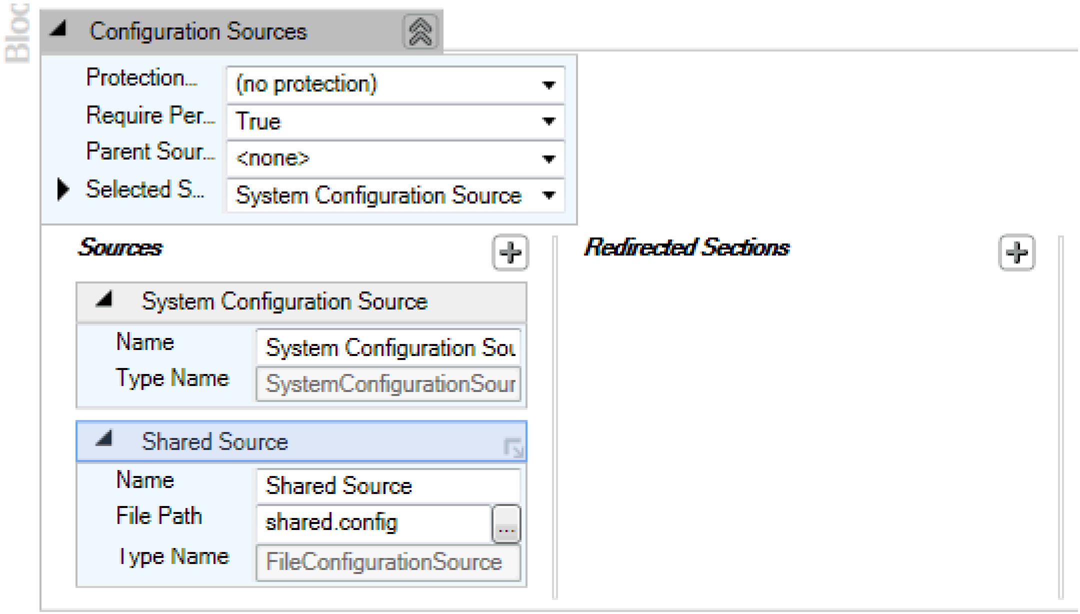
Redirected Sections

3. Click on the plus-sign icon next to **Sources**, select **Add Sources** from the context menu, and click **Add File-based Configuration Source.**

4. Set the following properties for the configuration element:

    ○ **Name**: Shared Source

    ○ **File Path**: shared.config



Configuration Sources

Protection...: (no protection)
Require Per...: True
Parent Sour...: <none>
Selected S...: System Configuration Source

Sources

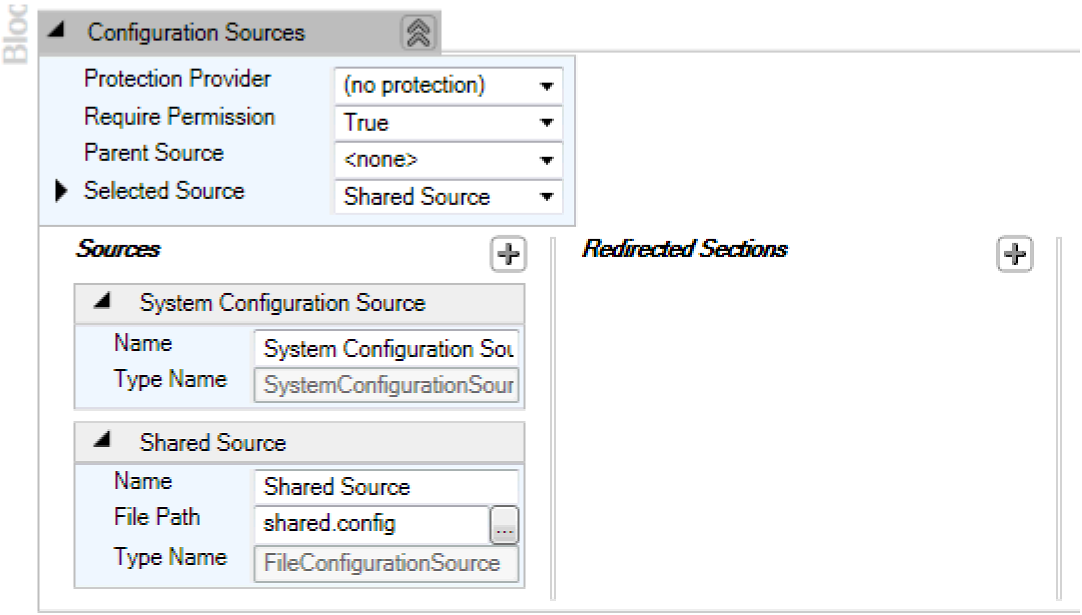
System Configuration Source
Name: System Configuration Sou
Type Name: SystemConfigurationSour

Shared Source
Name: Shared Source
File Path: shared.config
Type Name: FileConfigurationSource

Redirected Sections

5. In the properties for the **Configuration Sources** section, select **Shared Source** in the drop-down for **Selected Source**.

> You can resize the width of labels and values if you hover with your mouse over the left side of the text-boxes.



Protection Provider: (no protection)
Require Permission: True
Parent Source: <none>
Selected Source: Shared Source

System Configuration Source
Name: System Configuration Sou
Type Name: SystemConfigurationSour

Shared Source
Name: Shared Source
File Path: shared.config
Type Name: FileConfigurationSource

6. Right-click the **Configuration Sources** section and select **Validate**. If validation errors appear in the Visual Studio error list, correct the errors.

7. Save the configuration by using the **Save** command in Visual Studio, press Ctrl+S, or select **Save Puzzler1.config** from the **File** menu. The Enterprise Library configuration editor saves the Enterprise Library configuration to a file named **shared.config** in the directory where the **Puzzler1.config** file is located.

8. Close the Enterprise Library configuration editor.

9. In Visual Studio Solution Explorer locate the file **shared.config** by using either the **Show All Files** or **Refresh** button. Right-click the **shared.config** file and select **Include In Project**.

10. In the **Properties** window for **shared.config**, specify the following properties:

    - Build Action: **Content**

    - Copy to Output Directory: **Copy if newer**

When you run the application, the executable, **Puzzler1.config**, and **shared.config** will all be copied to the output folder. This is necessary for Enterprise Library to be able to locate the file **shared.config** because in step 4 you specified the file path as the relative path **shared.config**.

To set the Selected Source for Enterprise Library to an existing file

1. Open the **Puzzler2.config** file by double-clicking it inside the Solution Explorer.

> This will open the Visual Studio XML editor, not the Enterprise Library configuration editor.

> If you have previously set the Enterprise Library configuration editor to be the default editor for .config files, you can click **Open with…** and select **XML Editor** to open the configuration file.

2. In the Visual Studio XML editor, notice that the file contains an element called **<configSections>**. Add the following XML inside the **<configSections>** element.

   XML

   ```
   <section name="enterpriseLibrary.ConfigurationSource"
   type="Microsoft.Practices.EnterpriseLibrary.Common.Configuration.Configuratio
   nSourceSection, Microsoft.Practices.EnterpriseLibrary.Common,
   Version=5.0.414.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
   requirePermission="true" />
   ```

3. In the Visual Studio XML editor, after the closing tag **</configSections>** add the following XML:

   XML

   ```
     <enterpriseLibrary.ConfigurationSource selectedSource="Shared Source">
       <sources>
         <add name="System Configuration Source"
   type="Microsoft.Practices.EnterpriseLibrary.Common.Configuration.SystemConfig
   urationSource, Microsoft.Practices.EnterpriseLibrary.Common,
   Version=5.0.414.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" />

         <add name="Shared Source"
   type="Microsoft.Practices.EnterpriseLibrary.Common.Configuration.FileConfigur
   ationSource, Microsoft.Practices.EnterpriseLibrary.Common, Version=5.0.414.0,
   Culture=neutral, PublicKeyToken=31bf3856ad364e35"
           filePath="shared.config" />
       </sources>
     </enterpriseLibrary.ConfigurationSource>
   ```

4. Notice that the **selectedSource** attribute refers to the name of the <**add>** element, inside the <**sources>** element. Also notice that the **<add>** element has a **type** attribute that contains the type for **FileConfigurationSource** and the **filePath** attribute is set to **shared.config**.

5. Remove the following elements from within the **<configSections>** element:

   XML

   ```
       <section name="loggingConfiguration" .../>
       <section name="exceptionHandling" ... />
   ```

6. Remove the following elements from within the **<configuration>** element:

   XML

   ```
     <loggingConfiguration ... />
     <exceptionHandling ... />
   ```

7. Save and close the **Puzzler2.config** file.

To run the application

1. On the Visual Studio **Debug** menu select **Start Without Debugging** to run the application.
2. Notice that both applications behave consistently with regards to exception handling and logging.
3. Also notice that both applications have **shared.config** as their configuration file in the window's title.

Lab 2: Redirecting an Enterprise Library configuration section to a shared location

This lab demonstrates sharing a single Enterprise Library configuration section between separate applications (or application layers).

To begin this exercise, open the **Puzzler2.sln** file located in the ex02\begin folder.

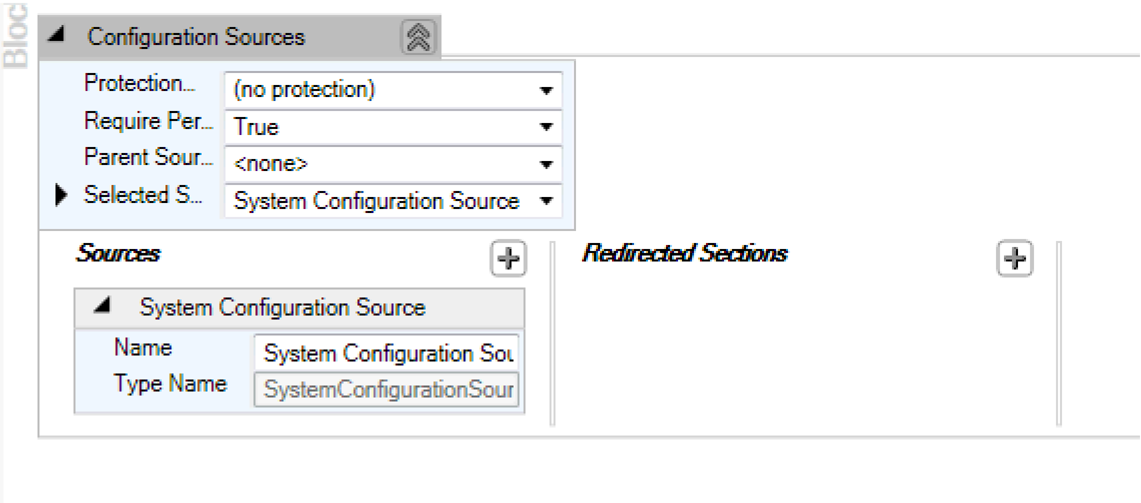
To review the application

This application behaves in the same way as the application that was used in Lab 1.

There is one additional file in the **PuzzlerUI** project called **shared.config**. This file contains a copy of the Logging Application Block settings used in the **Puzzler1.config** and **Puzzler2.config** files.

To redirect an Enterprise Library Configuration section

1. In the **PuzzlerUI** project, right-click the **Puzzler1.config** file and click **Edit Enterprise Library V5 Configuration**. This will open the Enterprise Library V5 configuration editor inside Visual Studio.
2. In the Enterprise Library configuration editor, on the **Blocks** menu, select **Add Configuration Settings**. This will add the **Configuration Sources** section to the configuration file. You can expand the section's settings by clicking on the arrows to the right of the text **Configuration Sources**.



Configuration Sources

Protection… (no protection)

Require Per… True

Parent Sour… <none>

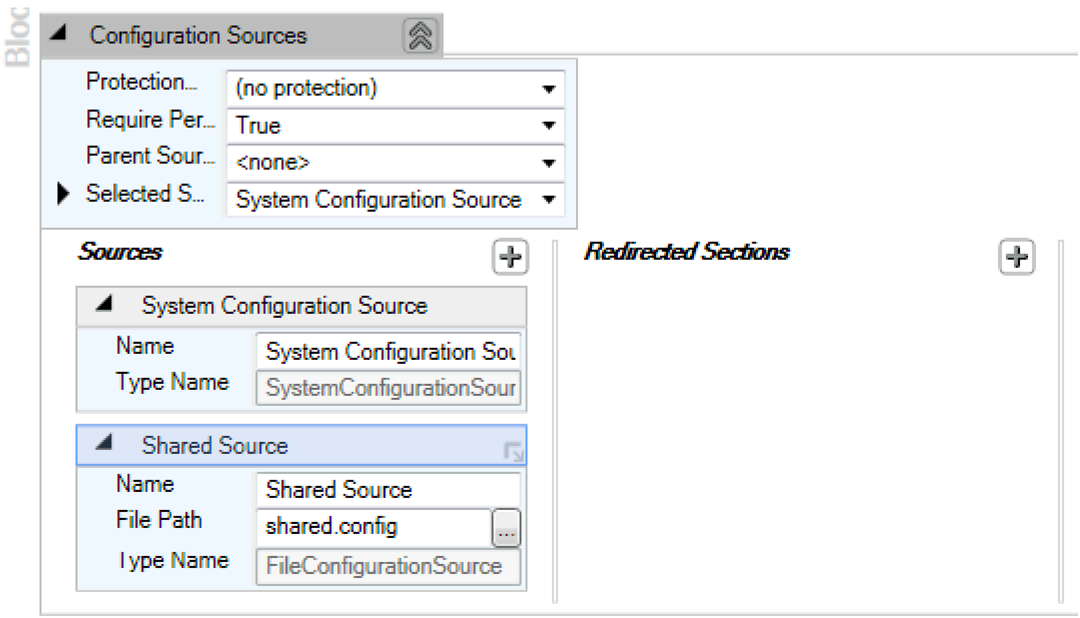
Selected S… System Configuration Source

System Configuration Source

Name: System Configuration Sou

Type Name: SystemConfigurationSour

3. Click on the plus-sign icon next to **Sources**, select **Add Sources** from the context menu, and click on **Add File-based Configuration Source.**
4. Set the following properties for the File-based Configuration Source:
   - **Name**: Shared Source
   - **File Path**: shared.config



Configuration Sources

Protection... (no protection)

Require Per... True

Parent Sour... <none>

Selected S... System Configuration Source

Sources

System Configuration Source

Name: System Configuration Sou

Type Name: SystemConfigurationSour

Shared Source

Name: Shared Source

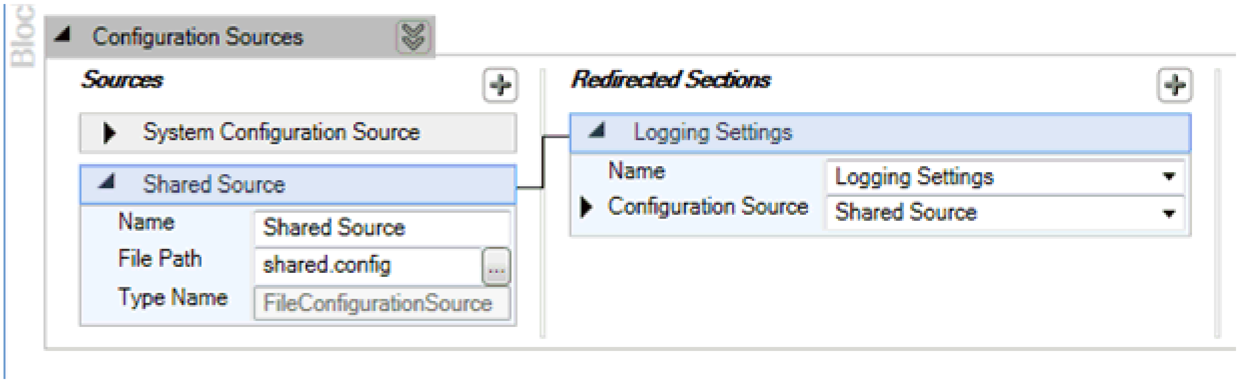
File Path: shared.config

Type Name: FileConfigurationSource

Redirected Sections

5. Click on the plus-sign icon next to **Redirected Section** and select **Add Redirected Section** from the context menu**.** Set the following properties for the redirected section:

   - **Name**: Logging Settings
   - **Configuration Source**: Shared Source



Configuration Sources

Sources

System Configuration Source

Shared Source

Name: Shared Source

File Path: shared.config

Type Name: FileConfigurationSource

Redirected Sections

Logging Settings

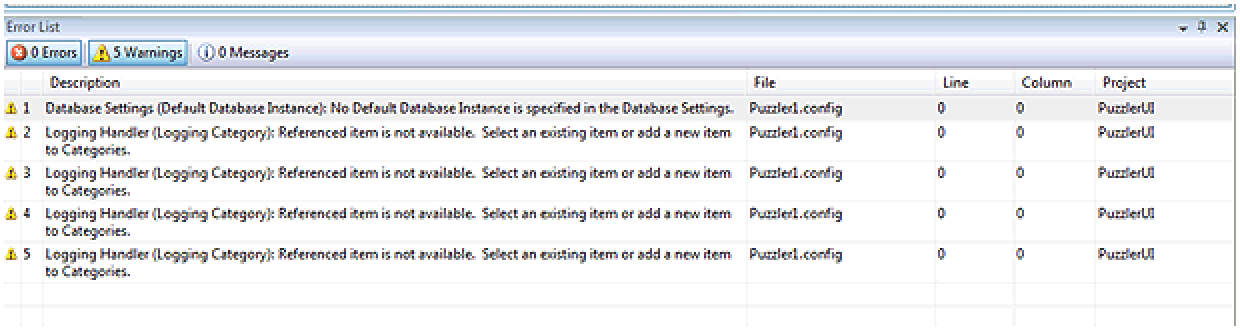
Name: Logging Settings

Configuration Source: Shared Source

6. Right-click the **Logging Settings** configuration section and choose **Delete Logging Settings.** When prompted to confirm the delete, select **Yes**.

7. Right-click the **Configuration Sources** section from the context menu and choose **Validate**.

8. Right-click the **Exception Handling Settings** section and select **Validate** from the **context menu**. In the Visual Studio error list a number of warnings will appear. These validation warnings indicate that settings from the logging section cannot be found. However at run time the settings will be retrieved from **shared.config** so you can ignore these warnings here.



	Description	File	Line	Column	Project
⚠ 1	Database Settings (Default Database Instance): No Default Database Instance is specified in the Database Settings.	Puzzler1.config	0	0	PuzzlerUI
⚠ 2	Logging Handler (Logging Category): Referenced item is not available. Select an existing item or add a new item to Categories.	Puzzler1.config	0	0	PuzzlerUI
⚠ 3	Logging Handler (Logging Category): Referenced item is not available. Select an existing item or add a new item to Categories.	Puzzler1.config	0	0	PuzzlerUI
⚠ 4	Logging Handler (Logging Category): Referenced item is not available. Select an existing item or add a new item to Categories.	Puzzler1.config	0	0	PuzzlerUI
⚠ 5	Logging Handler (Logging Category): Referenced item is not available. Select an existing item or add a new item to Categories.	Puzzler1.config	0	0	PuzzlerUI

9. Save the configuration using the save command in Visual Studio, use Ctrl+S, or on the configuration editor **File** menu select **Save Puzzler1.config.**
10. Repeat steps 1 through 9 for **Puzzler2.config**.

To run the application

1. On the Visual Studio **Debug** menu select **Start Without Debugging** to run the application.
2. Notice that both applications behave consistently with regards to exception handling and logging.

To verify that you have completed the exercise correctly, review the solution provided in the ex02\end folder.

Lab 3: Inheriting Enterprise Library configuration from a shared location

In this lab, you will inherit shared settings from an external location and override the configuration from the local (system configuration source) configuration file.

To begin this exercise, open the **Puzzler2.sln** file located in the ex03\begin folder.

To review the unencrypted configuration file

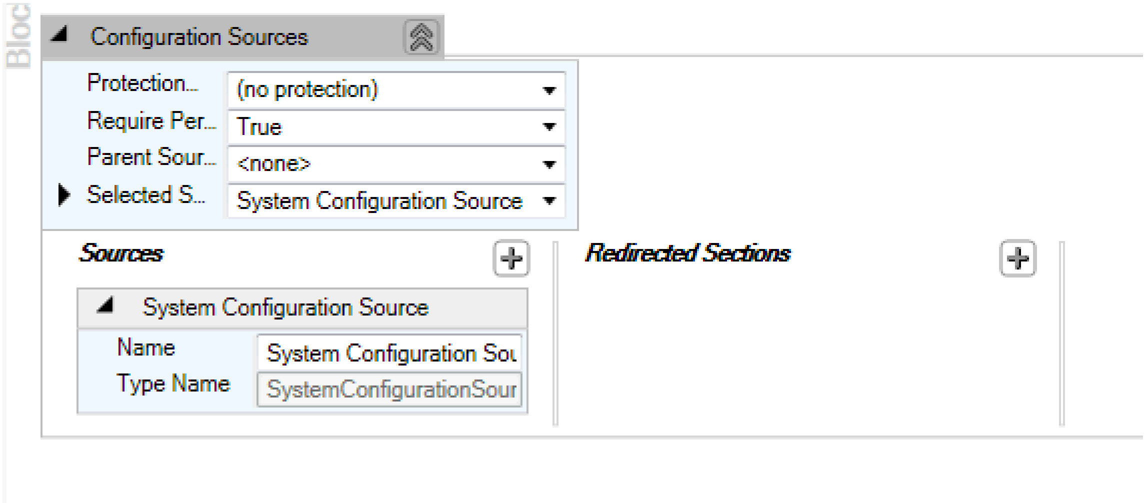
This application behaves in the same way as the application used in Lab 1.

There is one additional file in the **PuzzlerUI** project called **shared.config**. This file contains a copy of the Exception Handling Application Block settings used in the **Puzzler1.config** and **Puzzler2.config** files.

To specify a parent configuration source

1. In the **PuzzlerUI** project. Right-click the **Puzzler1.config** file and click the **Edit Enterprise Library V5 Configuration**. This will open the Enterprise Library V5 configuration editor inside Visual Studio.

2. In the Enterprise Library configuration editor, on the **Blocks** menu, click **Add Configuration Settings.** This will add the **Configuration Sources** section to the configuration file. You can expand the section's settings by clicking on the chevron icon to the right of the text **Configuration Sources**, as shown below.

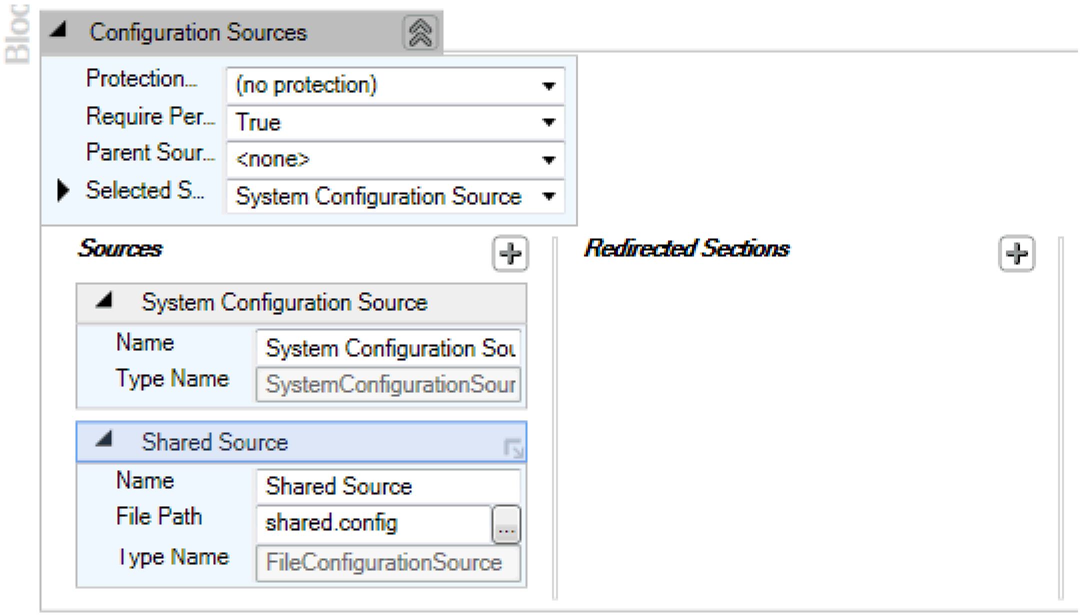


Protection... (no protection)
Require Per... True
Parent Sour... <none>
Selected S... System Configuration Source
Name System Configuration Sou
Type Name SystemConfigurationSour

3. Click on the plus-sign icon next to **Sources**, click on **Add Sources**, and then select **Add File-based Configuration Source.** Set the following properties for the File-based Configuration Source:

   - **Name**: Shared Source

   - **File Path**: shared.config



Configuration Sources

Protection...	(no protection)
Require Per...	True
Parent Sour...	<none>
Selected S...	System Configuration Source

System Configuration Source

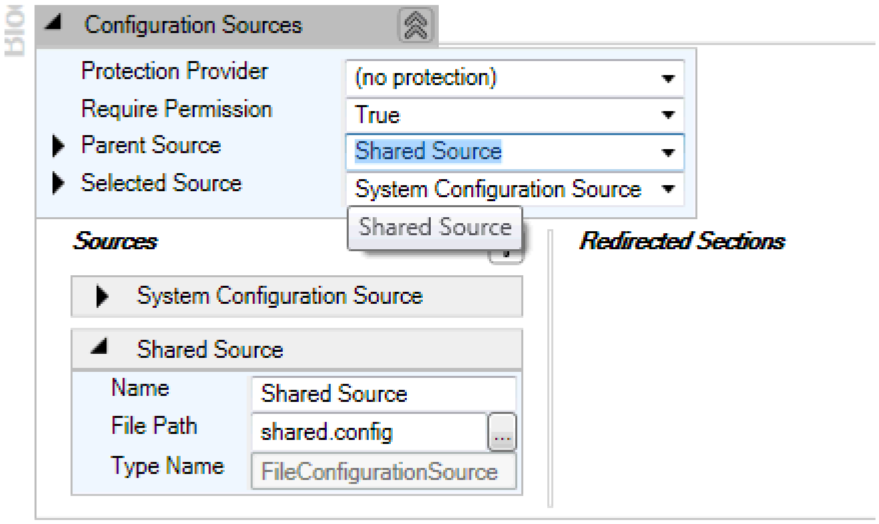
Name	System Configuration Sou
Type Name	SystemConfigurationSour

Shared Source

Name	Shared Source
File Path	shared.config
Type Name	FileConfigurationSource

4. In the properties for the **Configuration Sources** section, select **Shared Source** in the drop-down list for **Parent Source.**

> You can resize the width of labels and values if you hover with your mouse over the left side of the text-boxes.



Configuration Sources

Protection Provider	(no protection)
Require Permission	True
Parent Source	Shared Source
Selected Source	System Configuration Source

Shared Source

Name	Shared Source
File Path	shared.config
Type Name	FileConfigurationSource

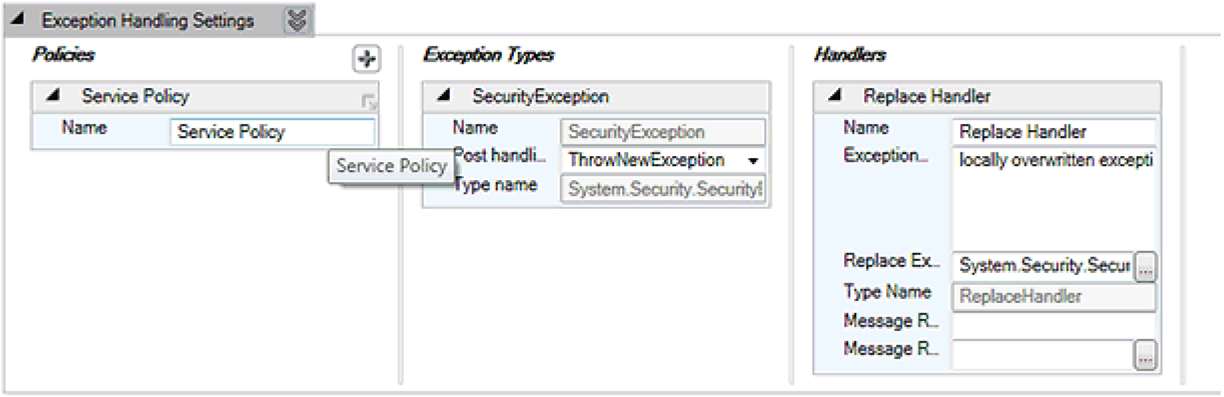
5. Right-click the **Exception Handling Settings** configuration section and click **Delete Exception Handling Settings**. When prompted to confirm the delete, select **Yes**.

6. Save the configuration by using the **Save** command in Visual Studio, press Ctrl+S, or select **Save Puzzler1.config** from the **File** menu.

7. Repeat steps 1 through 6 for **Puzzler2.config**.

To override inherited settings

1. In the **PuzzlerUI** project, right-click the **Puzzler2.config** file and select **Edit Enterprise Library V5 Configuration**. This will open the Enterprise Library V5 configuration editor inside Visual Studio.
2. On the Enterprise Library configuration editor **Blocks** menu select **Add Exception Handling Settings.**
3. Right-click the exception policy named **Policy** and select **Delete Policy**.
4. Click on the plus-sign icon next to **Policies**, click **Add Policy** from the context menu**.**
5. Set the value of the **Name** property to **Service Policy**.
6. Right-click the exception type named **All Exceptions** and select **Delete All Exceptions**.
7. Right-click the policy named **Service Policy**, then select **Add Exception Type** from the context menu. In the dialog that appears enter **SecurityException** as the type name and click **OK.**
8. Set the value of the Post handling action property to **ThrowNewException**.
9. Right-click the exception type named **SecurityException**, select **Add Handlers** from the context menu, and click **Add Replace Handler**.
10. Set the **Exception Message** property of the Replace Handler to **locally overwritten exception message**.
11. Click the ellipses (**...**) button in the **Replace Exception Type** property and select the **SecurityException** type in the type selector dialog.



Exception Handling Settings

Policies

Service Policy
Name: Service Policy

Exception Types

SecurityException
Name: SecurityException
Post handli..: ThrowNewException
Type name: System.Security.Security

Handlers

Replace Handler
Name: Replace Handler
Exception..: locally overwritten excepti
Replace Ex..: System.Security.Secur
Type Name: ReplaceHandler
Message R..:
Message R..:

The names and structure in steps 4 through 7 are very important. These must match the structure in **shared.config**. For the **Security Exception** type, you specified a **Replace Handler** with modified settings. The **Security Exception** does not contain a **Logging Handler**.

When the configuration is used at run time Enterprise Library's configuration source will **merge** the settings for **Puzzler2.config** with the settings in **shared.config**.
The Logging Handler for **Security Exception** will still apply in **Puzzler2.config**.
The **Replace Handler** will have a modified message.

12. Save the configuration using the **Save** command in Visual Studio, press Ctrl+S, or select **Save Puzzler2.config** from the **File** menu**.**

To run the application

1. On the Visual Studio **Debug** menu select **Start Without Debugging** to run the application.
2. Notice that the application configured with **Puzzler1.config** handles exceptions according to the settings specified in **shared.config**.
3. Notice that the application configured with **Puzzler2.config** handles exceptions consistently with **Puzzler1.config**, except for the **Replace Handler** on **SecurityException**. The **SecurityException** is still logged to the category **General**, as specified in **shared.config**.
4. You can force a **SecurityException** to occur by pressing the **Add Word** button for either application window.

To verify that you have completed the exercise correctly, review the solution provided in the ex03\end folder.

patterns & practices
proven practices for predictable results

Copyright