

You can replace DevForce logging with your own **custom logger**, or multiple loggers, to take full control over logging in your application.

The [ITraceLogger](#) is a simple interface DevForce will search for (via the [ITraceLoggerProvider](#)), and call the logger's *Log* method whenever a trace message should be logged. The *Log* method, although typed as receiving an *Object*, generally receives a [TraceMessage](#), which you can inspect to see where and when the message was generated and by whom.

The [debug log](#) is actually generated by an *ITraceLogger* implementation, the [TraceFileXmlLogger](#). If you don't provide your own *ITraceLogger* implementation, then it's the *TraceFileXmlLogger* which is used. If you do provide your own logger, the *TraceFileXmlLogger* will not be used.

In Silverlight, Windows Store apps and Azure, environments where a debug log is not written, a custom *ITraceLogger* is a good way of capturing trace and diagnostic information.

Here's a simple sample, which writes logging messages to the Console. Note that DevForce has a built-in *ConsoleLogger*, so if you do want to log to the Console you don't need a custom logger.

```
using System;
using IdeaBlade.Core;
namespace Samples {
    public class SampleLoggerProvider : ITraceLoggerProvider {
        public ITraceLogger GetLogger() {
            return new SampleConsoleLogger();
        }
    }
    // Sample logger - note that DevForce supplies a class named ConsoleLogger - this is shown here
    // as a sample only.
    public class SampleConsoleLogger : ITraceLogger {
        /// <summary>
        /// Return true if your logger is not thread-safe.
        /// </summary>
        public bool IsSingleThreaded {
            get { return false; }
        }
        /// <summary>
        /// Log the trace message - here we write to the Console.
        /// </summary>
        /// <param name="message"></param>
        public void Log(object message) {
            Console.WriteLine(message);
        }
    }
}
```

```
Imports System
Imports IdeaBlade.Core
Namespace Samples
    Public Class SampleLoggerProvider
        Implements ITraceLoggerProvider
        Public Function GetLogger() As ITraceLogger
            Return New SampleConsoleLogger()
        End Function
    End Class
    ' Sample logger - note that DevForce supplies a class named ConsoleLogger - this is shown here
    ' as a sample only.
    Public Class SampleConsoleLogger
        Implements ITraceLogger
        "" <summary>
        "" Return true if your logger is not thread-safe.
        "" </summary>
        Public ReadOnly Property IsSingleThreaded() As Boolean
        Get
            Return False
        End Get
        End Property
        "" <summary>
        "" Log the trace message - here we write to the Console.
        "" </summary>
        "" <param name="message"></param>
        Public Sub Log(ByVal message As Object)
            Console.WriteLine(message)
        End Sub
    End Class
End Namespace
```

```
End Class  
End Namespace
```