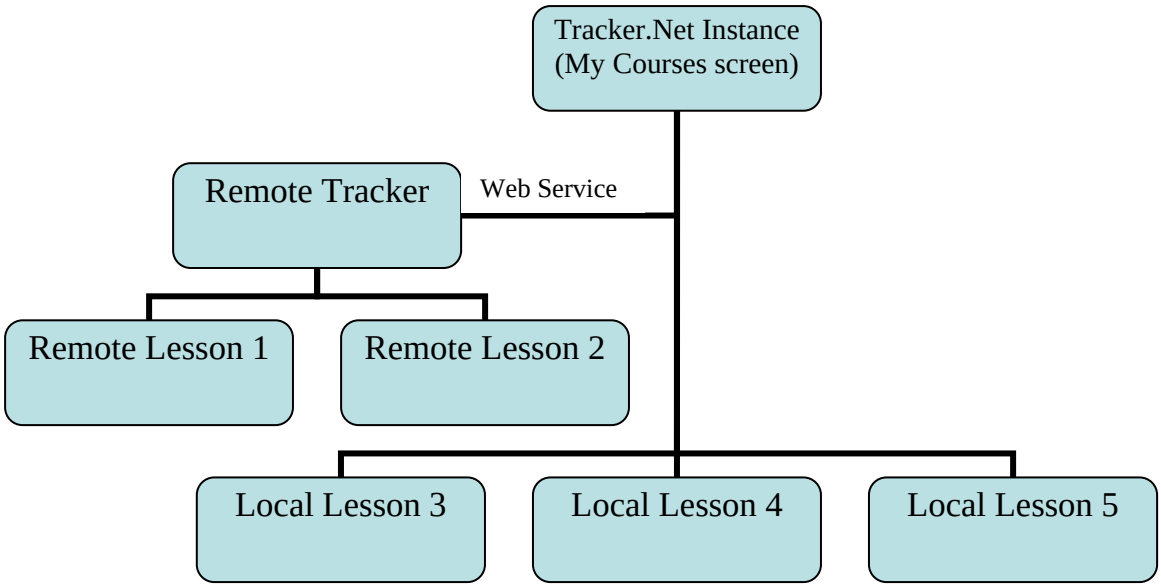
Remote Tracker Documentation

Table of Contents

This document describes the Remote Tracker configuration and implementation. Remote Tracker allows you to locate your SCORM content in a different domain than your Tracker.Net instance. It is needed because security restrictions in all browsers prevent cross-site JavaScript calls. Since the SCORM Runtime Environment uses JavaScript to communicate between the content (Sharable Content Object [SCO]) and the Learning Management System (LMS), the content and LMS typically need to be in the same domain (URL). Remote Tracker gets around this restriction.

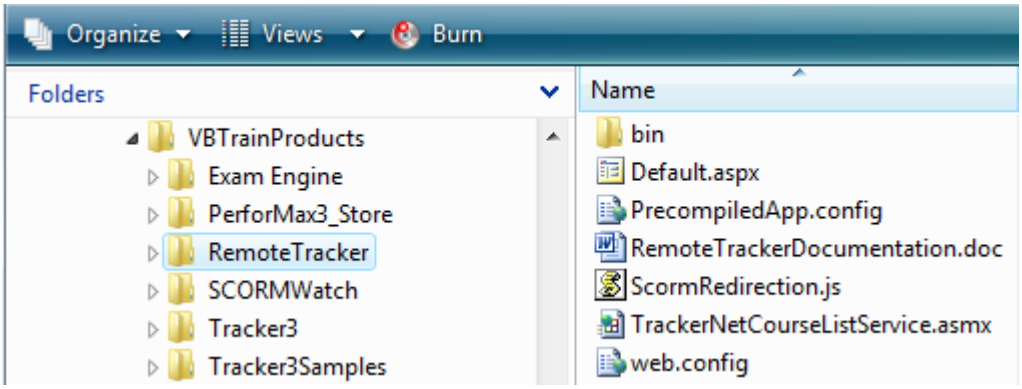
Design

Remote Tracker works by in essence putting a local copy of an LMS on the remote site and then communicating the data back to your Tracker.Net instance via a web service. This is shown in the diagram below.



Directory Structure

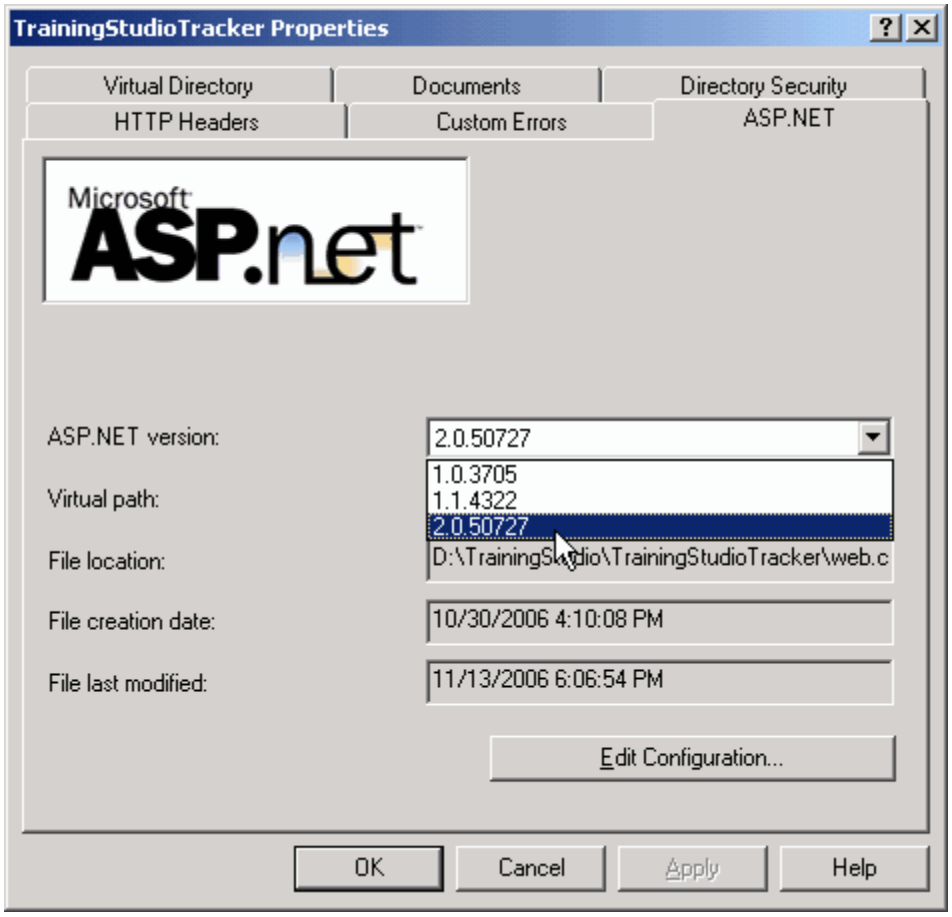
The Remote Tracker web service needs to be configured on the remote web site. Since it is an ASP.NET 2.0 application, the remote web site must be a Windows computer with the free ASP.NET 2.0 runtime installed. The layout of the web service directory structure is shown to the right. You would normally copy the entire *RemoteTracker* directory structure to your web server and then configure it as described in the next section. The *Default.aspx* file is the one you actually link to from Tracker.Net, passing the URL to the actual content as part of the query string.



The *web.config* file has a number of user-configurable settings as discussed later in this document. The rest of the files are used internally by the Remote Tracker application.

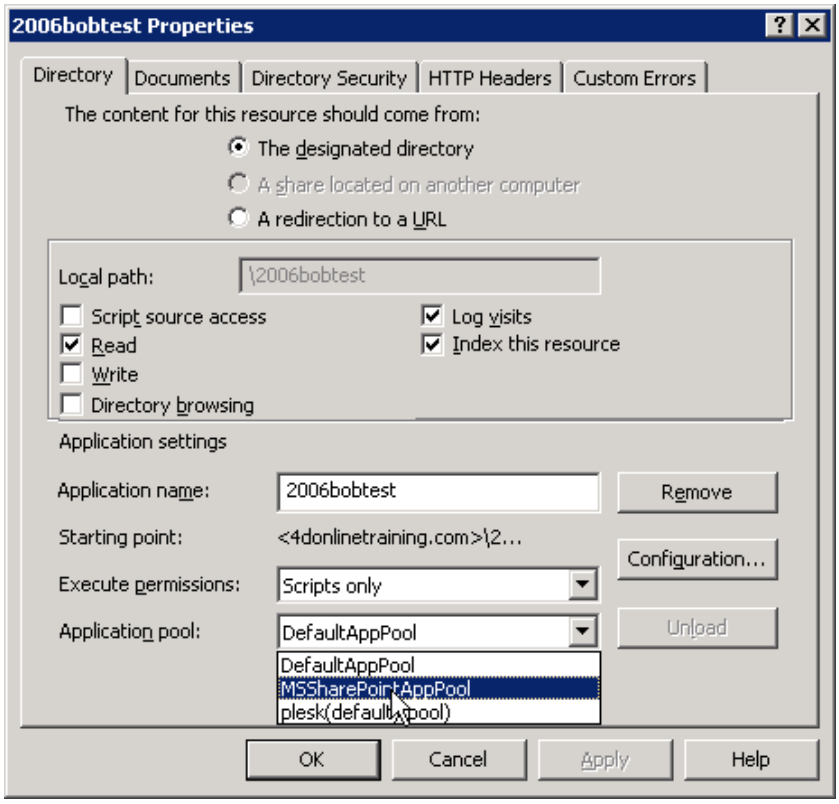
Deploying the Application

To begin working with the application, you must put its directory structure on the web server and make it into an ASP.NET 2.0 virtual directory/application. If your local computer has Internet Information Services (IIS) and ASP.NET installed, you can do this locally for testing as well. If you have multiple versions of the .NET runtime on the web server, then you will be able to choose 2.0 from the ASP.NET tab as shown below. Note that this screen capture is for a different virtual directory name – yours would be *RemoteTracker* but could be another virtual directory name. You might choose another name if you are hosting multiple web services on your site for different customers.

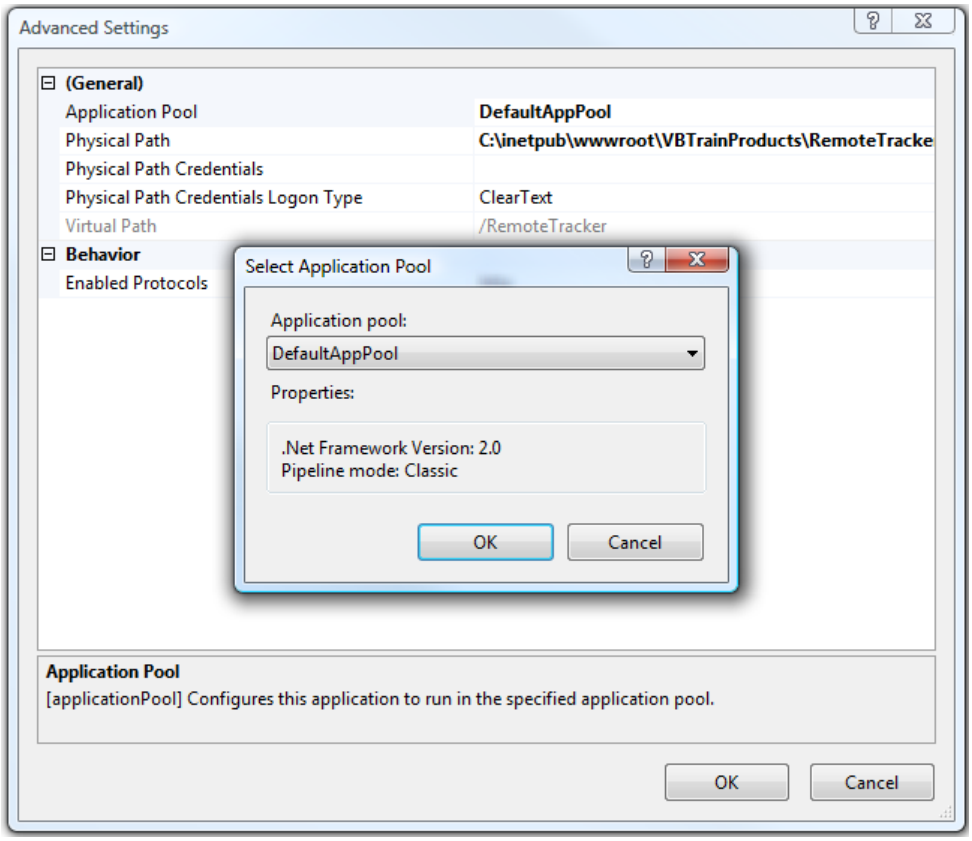


Note: If you are running IIS 6 or later, it is critical that you select an application pool that only has ASP.NET 2.0 applications in it.

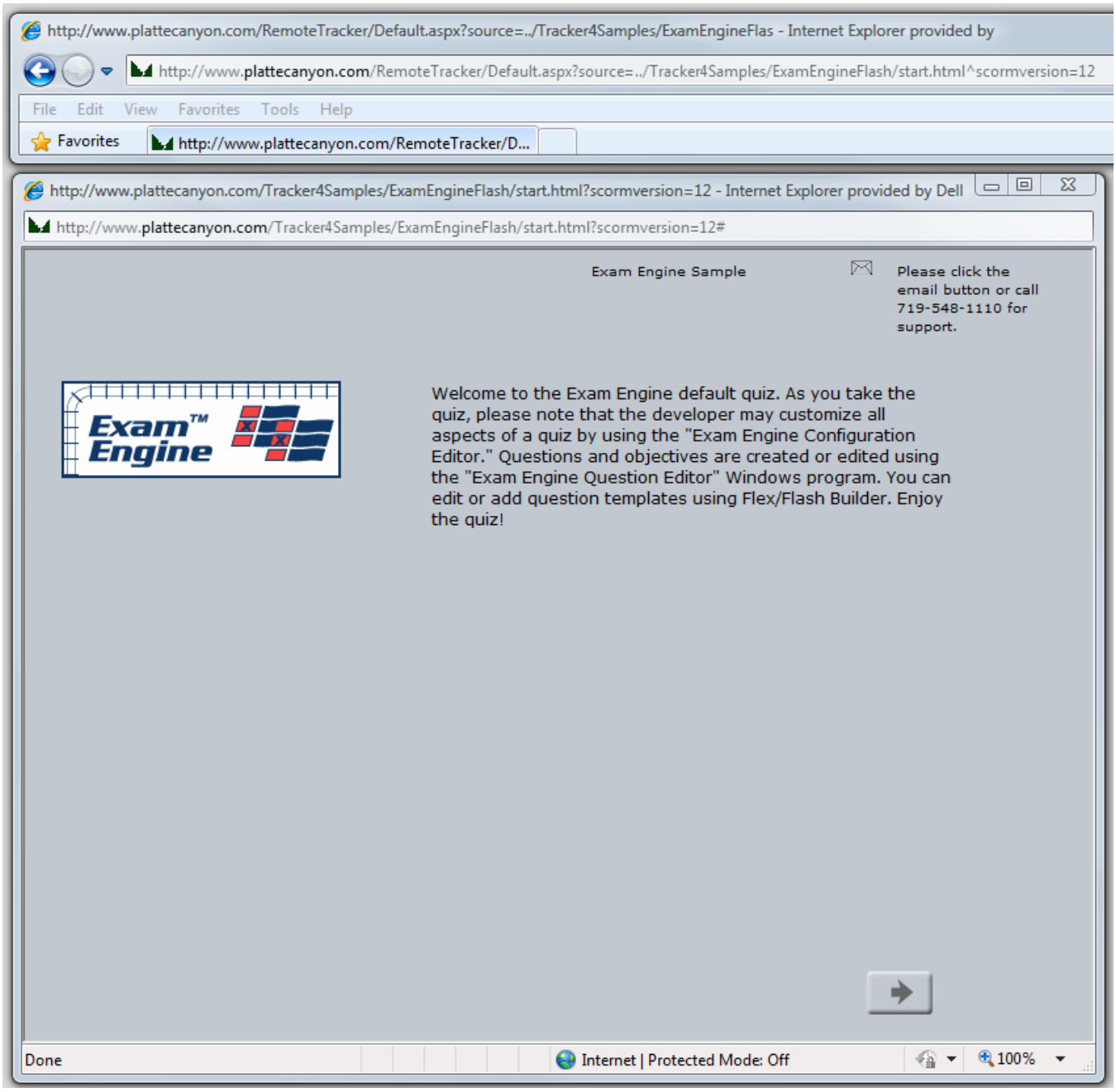
Mixing .NET 1.1 and 2.0 applications within the same pool can cause both the web service and the 1.1 application to fail. The setting for an application pool is shown below.


2006bobtest Properties
Directory
Documents
Directory Security
HTTP Headers
Custom Errors
The content for this resource should come from:
The designated directory
A share located on another computer
A redirection to a URL
Local path:
\2006bobtest
Script source access
Read
Write
Directory browsing
Log visits
Index this resource
Application settings
Application name:
2006bobtest
Remove
Starting point:
<4donlinetraining.com>\2...
Configuration...
Execute permissions:
Scripts only
Application pool:
DefaultAppPool
Unload
DefaultAppPool
MSSharePointAppPool
plesk(default pool)
OK
Cancel
Apply
Help

Or


Advanced Settings
(General)
Application Pool
DefaultAppPool
Physical Path
C:\inetpub\wwwroot\VBTrainProducts\RemoteTracke
Physical Path Credentials
Physical Path Credentials Logon Type
ClearText
Virtual Path
/RemoteTracker
Behavior
Enabled Protocols
Select Application Pool
Application pool:
DefaultAppPool
Properties:
.Net Framework Version: 2.0
Pipeline mode: Classic
OK
Cancel
Application Pool
[applicationPool] Configures this application to run in the specified application pool.
OK
Cancel

You can tell if the application is set up correctly if you can navigate to the *Default.aspx* file with a correct URL and get a popup screen like the one shown.



As discussed later in this document, you pass in the actual URL to the content using this syntax:

?source=../DirectoryName/startingPage.html

If you would prefer not to the use the ../ (which means go up one level from where you are), you can put in a complete URL. Note that this URL must be the same domain as Remote Tracker instance or

otherwise we run into the same cross-domain problem that we are trying to solve. So you could use this syntax as well:

?source=http://www.yoururl.com/DirectoryName/startingPage.html

If your URL has its own query string parameters, use a ^ symbol for ? and the | symbol for &.

Editing the Remote Tracker web.config File

The web.config file located in the main directory of the application has three user-configurable settings. These are located in the <*appSettings*> area of the file. Each setting is described below. After editing the web.config (and keeping a backup copy of the original), you can replace the one on your web server.

1. **TrackerScormWrapperAccessKey**. This key must match the key defined in web.config file of your Tracker.Net instance. This is discussed in the next section.

```
<add key="TrackerScormWrapperAccessKey" value="1111111111" />
```

2. **Tracker.TrackerNetUtilityService**. This key is the URL to the web service within the Tracker.Net instance you are communicating with. Note that it must point to the *TrackerNetUtilityService.asmx* file located in the *NewStudent* directory within this instance.

```
<add key="Tracker.TrackerNetUtilityService"
value="http://www.plattecanyon.com/tracker4/NewStudent/TrackerNetUtil
ityService.asmx"/>
```

3. **ExternalStyleSettings**. By default, the Remote Tracker application launches the content in a separate window[1]. This is needed to capture the end of the session when the user clicks the "x" on the browser window or presses the Alt + F4 keyboard combination. This key controls the characteristics of this window other than the width and height (which we capture from what you set in Tracker.Net). The values are likely self-explanatory. They are the left and top position in pixels, whether there is a status bar, whether the window is resizable, whether the window will have scrollbars if needed, whether it will have a toolbar, and whether it will have a menubar.

```
<add key="ExternalStyleSettings"
value="left=0,top=0,status=no,resizable=yes,scrollbars=yes,toolbar=no
,menubar=no"/>
```

Editing the Tracker.Net web.config File

The web.config file located in the main directory of the Tracker.Net instance also has numerous user-configurable settings. These are located in the <*appSettings*> area of the file. If the TrackerScormWrapperAccessKey key is not already defined, add it to your existing file. After editing the web.config (and keeping a backup copy of the original), you can replace the one on your web server.

[1] You can make it show in the original window by adding &new=false to the URL. However, this will result in no SCORM message when the user closes the browser window in most authoring tools.

1. **TrackerScormWrapperAccessKey**. This key must match the key defined in web.config file of the Remote Tracker instance(s). This is discussed in the previous section.

```
<add key="TrackerScormWrapperAccessKey" value="1111111111" />
```

Linking to a Remote Lesson

You use the *external* keyword to distinguish a remote lesson from a normal lesson in Tracker.Net. You add remote lessons in the same way as normal lessons (Administrator – Lessons screen). You then use this syntax for the remote lesson:

external=http://www.yoururl.com/RemoteTracker/Default.aspx?source=<path to actual lesson>

Here are some examples:

external=http://www.plattecanyon.com/RemoteTracker/Default.aspx?source=../Tracker4Samples/ToolBook10_13/index.html

external=http://www.plattecanyon.com/RemoteTracker/Default.aspx?source=../Tracker4Samples/ExamEngine3Flash/start.html^scormversion=12

As mentioned above, notice how we replace the ? with ^ when the lesson itself has a query string parameter. If there are additional query string parameters, replace the & with |. So if you needed http://www.plattecanyon.com/Tracker4Samples/ExamEngineFlash/start.html?scormversion=12&title=XYZ, then you would use: http://www.plattecanyon.com/Tracker4Samples/ExamEngineFlash/start.html^scormversion=12|title=XYZ.

Here is how a lesson location looks in Tracker.Net.

By default, the lesson will pop up in its own window. The original window will then shrink to a size of 10 x 10 pixels. Popping up a window is most reliable since many authoring tools capture the window closing event in order to send the proper *LMSFinish/Terminate* messages[2]. If this is not a problem for your content, you can add the *&new=false* query string to the end of the lesson location.

Configuring TrackerCourseList.js

By default, the *My Courses* screen within Tracker.Net checks for remote lesson completion every 5 seconds. That means there is a slight delay between the time that the user finishes the lesson and the screen updates. If the user tries to open another lesson in the meantime, the screen checks immediately for completion. You can control the time by changing this line in the *scripts\TrackerCourseList.js* file:

```
var timerLength = 5000; // timer interval in milliseconds for remote
                        // lesson to check for status
```

[2] If we don't pop up in another window, Remote Tracker will show the lesson in an *iFrame* control. Unfortunately, the closing of the overall window does not properly trigger the window closing event for the page shown in the *iFrame*.

The trade-off is that setting this shorter will put a bit more load on the system.

Configuring ScormRedirection.js

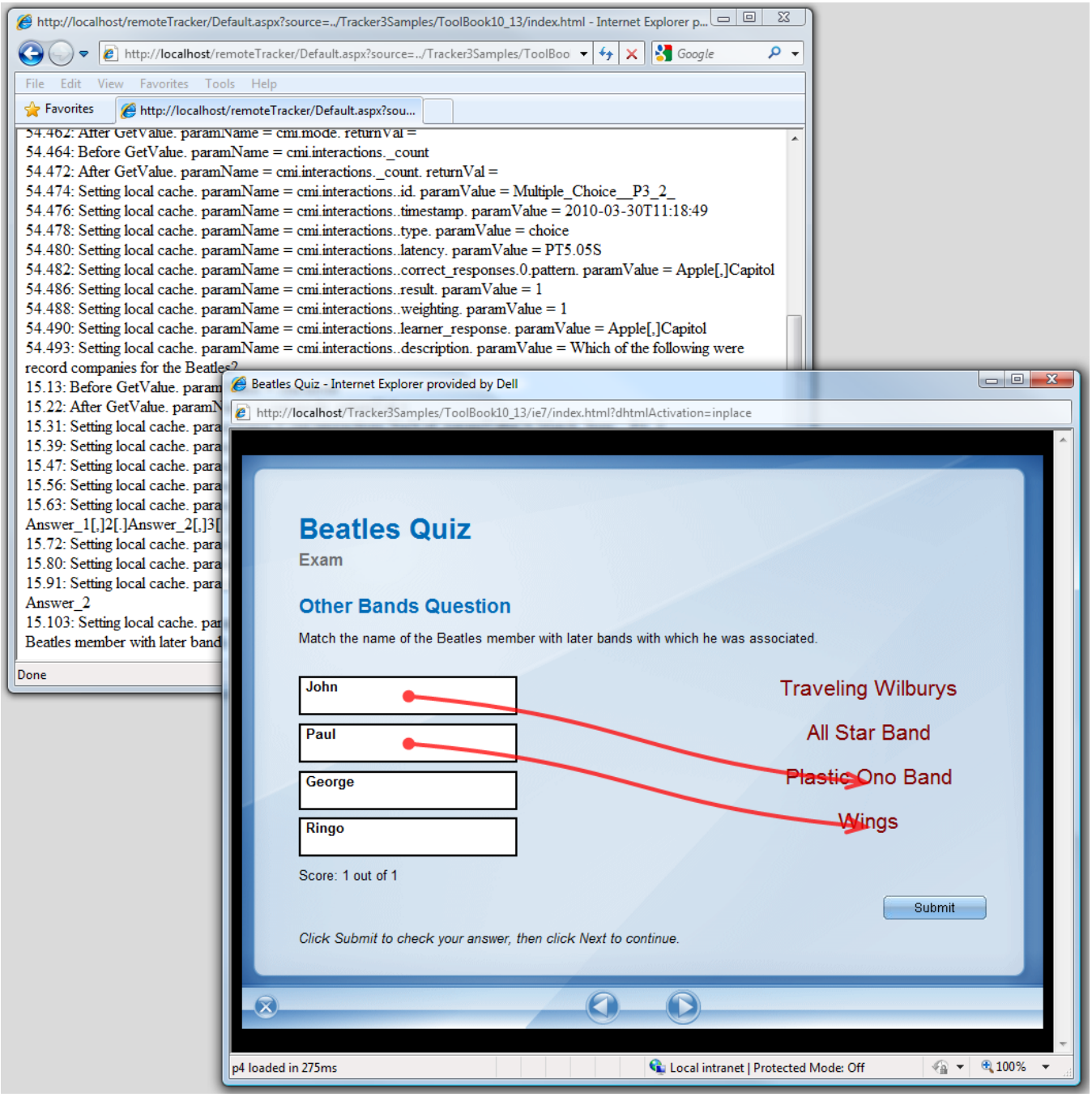
If you are having any issues with Remote Tracker or if you would just like to verify the SCORM messages, you can turn on debugging within the *ScormRedirection.js* file with the Remote Tracker application. Change this line:

```
var debug = false;
```

to

```
var debug = true;
```

This will result in the original window displaying all the SCORM messages. You will need to resize this window and possibly scroll down to see all the messages. This is shown below:


http://localhost/remoteTracker/Default.aspx?source=../Tracker3Samples/ToolBook10_13/index.html - Internet Explorer p...
File Edit View Favorites Tools Help
Favorites
54.464: Before GetValue. paramName = cmi.interactions._count
54.472: After GetValue. paramName = cmi.interactions._count. returnVal =
54.474: Setting local cache. paramName = cmi.interactions..id. paramValue = Multiple_Choice__P3_2_
54.476: Setting local cache. paramName = cmi.interactions..timestamp. paramValue = 2010-03-30T11:18:49
54.478: Setting local cache. paramName = cmi.interactions..type. paramValue = choice
54.480: Setting local cache. paramName = cmi.interactions..latency. paramValue = PT5.05S
54.482: Setting local cache. paramName = cmi.interactions..correct_responses.0.pattern. paramValue = Apple[,]Capitol
54.486: Setting local cache. paramName = cmi.interactions..result. paramValue = 1
54.488: Setting local cache. paramName = cmi.interactions..weighting. paramValue = 1
54.490: Setting local cache. paramName = cmi.interactions..learner_response. paramValue = Apple[,]Capitol
54.493: Setting local cache. paramName = cmi.interactions..description. paramValue = Which of the following were
record companies for the Beatles?
Done
Beatles Quiz - Internet Explorer provided by Dell
http://localhost/Tracker3Samples/ToolBook10_13/ie7/index.html?dhtmlActivation=inplace
Beatles Quiz
Exam
Other Bands Question
Match the name of the Beatles member with later bands with which he was associated.
John
Paul
George
Ringo
Traveling Wilburys
All Star Band
Plastic Ono Band
Wings
Score: 1 out of 1
Submit
Click Submit to check your answer, then click Next to continue.
p4 loaded in 275ms
Local intranet | Protected Mode: Off
100%