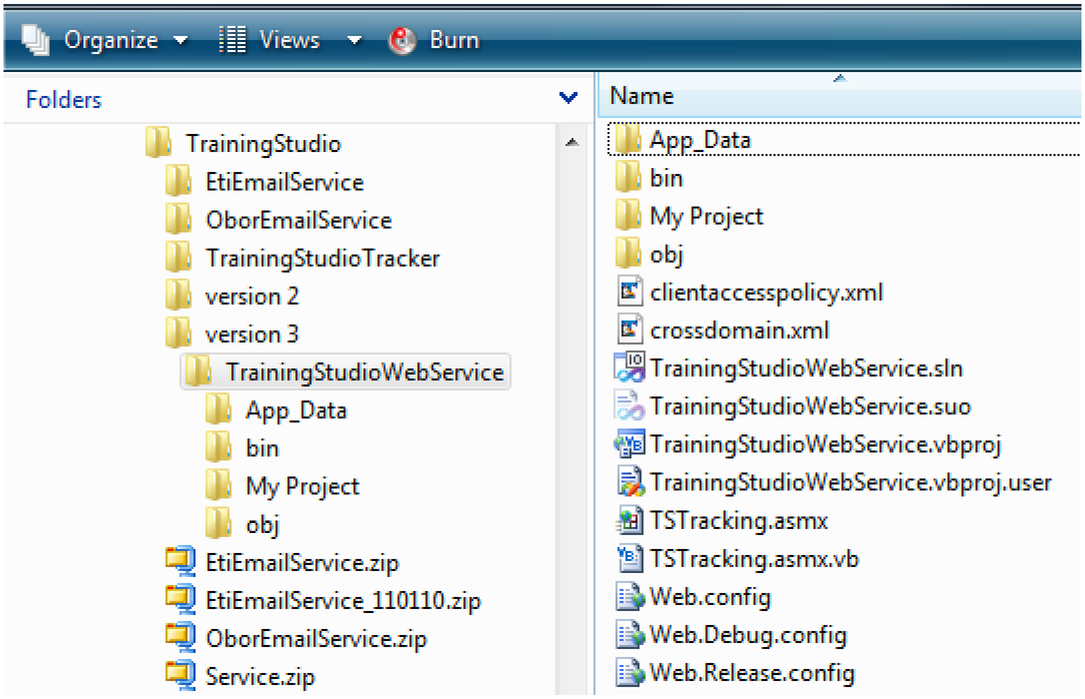
Training Studio Web Service Documentation

Table of Contents

This document describes the Training Studio Web Service configuration and implementation. Note that the use of this web service is optional. It is used to provide these pieces of Training Studio functionality:

- Email technical support.
- Save reviewer comments
- The ability to run the *Training Studio Tracker Administrator* to view, edit, and close comments.

Directory Structure

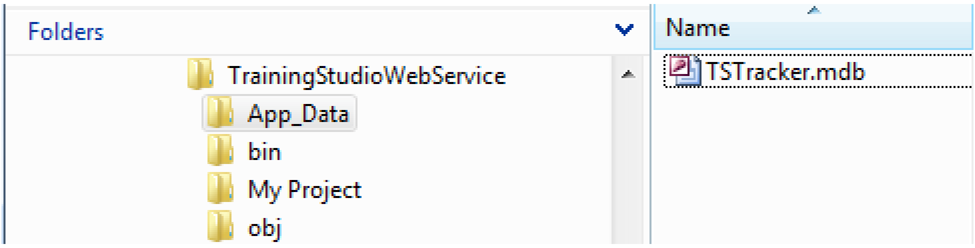


TrainingStudio
EtiEmailService
OborEmailService
TrainingStudioTracker
version 2
version 3
TrainingStudioWebService
App_Data
bin
My Project
obj
EtiEmailService.zip
EtiEmailService_110110.zip
OborEmailService.zip
Service.zip

App_Data
bin
My Project
obj
clientaccesspolicy.xml
crossdomain.xml
TrainingStudioWebService.sln
TrainingStudioWebService.suo
TrainingStudioWebService.vbproj
TrainingStudioWebService.vbproj.user
TSTracking.asmx
TSTracking.asmx.vb
Web.config
Web.Debug.config
Web.Release.config

The layout of the web service directory structure is shown to the right. You would normally copy the entire *TrainingStudioWebService* directory structure to your web server and then configure it as described in the next section. The *TSTracking.asmx* file is the actual web service referenced by the "support email" functionality, the "comments" capability within the training, and by the *Training Studio Tracker Administrator* application. The *TrainingStudioWebService.sln* is a Visual Studio .NET 2010 solution file for editing the web service program. The *crossdomain.xml* and *clientaccesspolicy.xml* files are special XML file that allows client-side applications access the service. They are not needed by this version of Training Studio, but it is a good idea to keep them around just in case. The *Web.config* file has a number of user-configurable settings as discussed later in this document.

The *TSTracking.asmx.vb* file is a code file that contains the programming of the web service. It should not be edited directly but rather should be maintained through editing the solution via Visual Studio .NET 2010.



TrainingStudioWebService
App_Data
bin
My Project
obj

TSTracker.mdb

The *App_Data* directory contains the *TSTracker.mdb* Access database as shown to the right. This database contains the list of categories to use as well as the actual comments, reviewers, and administrators created via the *Comments* feature within the training and the *Training Studio Tracker Administrator* application. The editable tables of this database are covered later in this document.

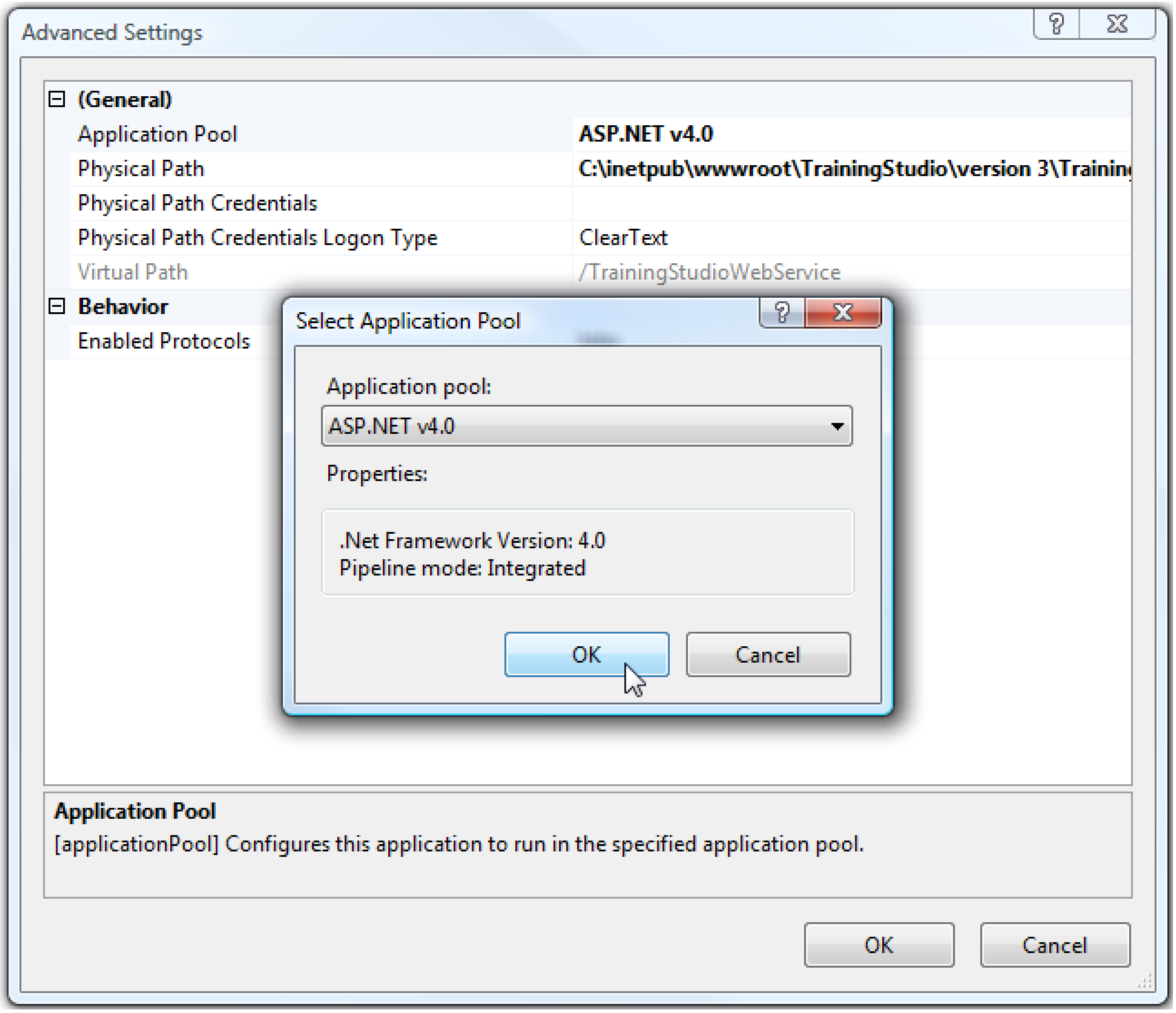
Deploying the Web Service

To begin working with the web service, you must put its directory structure on the web server and make it into an ASP.NET 4.0 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 are running IIS 6, you select the ASP.NET version via the ASP.NET tab.

Note: If you are running IIS 6 or later, it is critical that you select an application pool that only has ASP.NET 4.0 applications in it.

Mixing applications with different .NET versions within the same pool can cause both the web service and the other applications to fail.

For IIS 7 and later, you pick the ASP.NET version via the application pool. This is shown below.



Advanced Settings

(General)

Application Pool	ASP.NET v4.0
Physical Path	C:\inetpub\wwwroot\TrainingStudio\version 3\Trainin
Physical Path Credentials	
Physical Path Credentials Logon Type	ClearText
Virtual Path	/TrainingStudioWebService

Behavior

Enabled Protocols

Select Application Pool

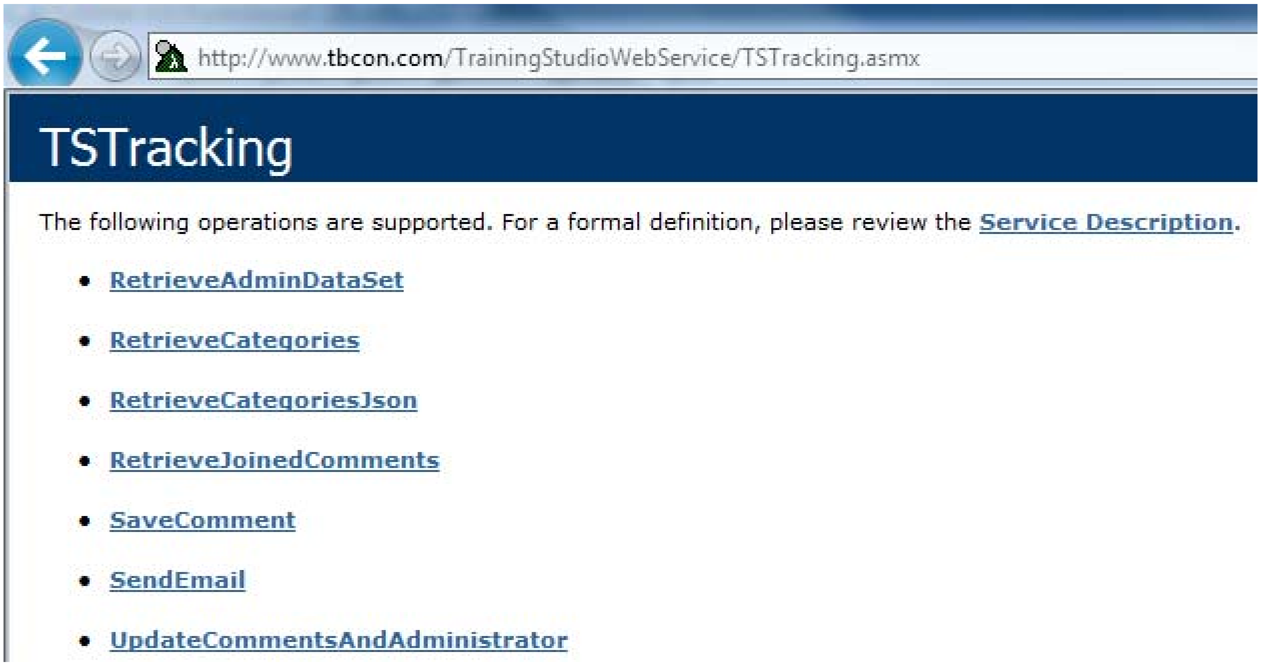
Application pool: ASP.NET v4.0

Properties:

.Net Framework Version: 4.0
Pipeline mode: Integrated

Application Pool
[applicationPool] Configures this application to run in the specified application pool.

You can tell if the web service is set up correctly if you can navigate to the *TSTracking.asmx* file and see a screen like the one below.



http://www.tbcon.com/TrainingStudioWebService/TSTracking.asmx

TSTracking

The following operations are supported. For a formal definition, please review the Service Description.

- RetrieveAdminDataSet
- RetrieveCategories
- RetrieveCategoriesJson
- RetrieveJoinedComments
- SaveComment
- SendEmail
- UpdateCommentsAndAdministrator

Note that this URL is the *WebServiceUrl* set in the *Training Structure* using the *Training Studio Content Editor*. It is also needed by the users of the *Training Studio Tracker Administrator* application. If you are prompted for any type of logon before reaching this screen, then neither the support email nor *Comments* functionality will work.

Due to security restrictions for JavaScript, you normally want to locate the web service in the same domain as your content. For example, if your content is on www.tbcon.com, you want the web service on that URL as well, such as www.tbcon.com/TrainingStudioWebService. Exam Engine attempts to make a cross-domain call if needed, but this tends to work only with Internet Explorer. If you need to go cross-domain, we recommend that you place the provided *crossdomain.xml* file on the web server containing the web service. Although primarily designed for Silverlight content, you might also include *clientaccesspolicy.xml* file in the same location. It normally needs to be set on the *root* of the web site. For example, we place it at http://www.plattecanyon.com rather than at http://www.plattecanyon.com/TrainingStudioWebService. Here are the contents of these files:

crossdomain.xml

```
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy SYSTEM
"http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd">
<cross-domain-policy>
     <site-control permitted-cross-domain-policies="all"/>
     <allow-access-from domain="*" />
     <allow-http-request-headers-from domain="*" headers="SOAPAction"
/>
</cross-domain-policy>
```

clientaccesspolicy.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<access-policy>
     <cross-domain-access>
          <policy>
               <allow-from http-request-headers="*">
                    <domain uri="*"/>
               </allow-from>
               <grant-to>
                    <resource include-subpaths="true" path="/"/>
               </grant-to>
          </policy>
     </cross-domain-access>
</access-policy>
```

These files allow communications from any domain. You can make these files more restrictive if desired. See http://www.adobe.com/devnet/articles/crossdomain_policy_file_spec.html for information on *crossdomain.xml* and http://msdn.microsoft.com/en-us/library/cc197955(VS.95).aspx for information on *clientaccesspolicy.xml*.

Editing the Web.config File

The Web.config file located in the main directory of the web service has a number of user-configurable settings. These are all 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. **ReviewTrackingKey**. This key must match the key defined in the TrainingStudioSupport.js file (*reviewKey* variable) within your Training Studio application. You may want to set up different web services (and matching TrainingStudioSupport.js files) for different customers or subject areas.

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

2. **AdminTrackingKey**. This key must match the key being sent by the *Training Studio Tracker Administrator* application as described later in this document. This key gives the Administrator the right to view the comments, make changes to their status, etc.

```
<add key="AdminTrackingKey" value="222222222" />
```

3. **SendEmailNotificationOfComment**. This setting controls whether the web service sends an email when it receives a comment. It can have a value of True or False.

```
<add key="SendEmailNotificationOfComment" value="True" />
```

4. **NotificationEmailAddress**. This is the email address(es) to which notification emails are sent. Separate multiple emails by a comma. This setting is only relevant if *SendEmailNotificationOfComment* is true.

```
<add key="NotificationEmailAddress" value="jrhodes@plattecanyon.com"
/>
```

5. **NotificationEmailMessage**. This is the body of the notification email message. There are placeholders for variable pieces of information as follows.

Placeholder	Value
{0}	Carriage Return Line Feed (e.g., a hard return)
{1}	Application title
{2}	Page number of the training that the comment refers to
{3}	First name of reviewer
{4}	Last name of reviewer
{5}	Organization of reviewer
{6}	Contact phone number of reviewer
{7}	Email address of reviewer
{8}	Description of the issue referenced in the comment
{9}	Steps to recreate the issue described in the comment

This setting is only relevant if *SendEmailNotificationOfComment* is true.

```
<add key="NotificationEmailMessage" value="Dear
Administrator,{0}{0}There is a comment posted for application {1}
from {3} {4} of {5}. The comment is on page {2}. The description is
'{8}' and the recreation steps are '{9}'. {0}{0}Sincerely,{0}{0}The
Training Studio Tracker Team" />
```

6. **NotificationEmailSubject**. This is the subject line of the notification email. This setting is only relevant if *SendEmailNotificationOfComment* is true.

```
<add key="NotificationEmailSubject" value=" Training Studio Tracker
Information" />
```

7. **NotificationEmailReplyAddress**. This is the reply address of the notification email. This setting is only relevant if *SendEmailNotificationOfComment* is true.

```
<add key="NotificationEmailReplyAddress"
value="support@plattecanyon.com" />
```

8. **SmtpServer**. This is the Simple Mail Transfer Protocol (SMTP) server for the notification email. This will be specific to your web server. If you are running the web service on your local

computer for testing, you would put the machine name in for this setting. You can get the machine name via *My Computer – Properties*. Using the machine name or “(local”) works for some web servers as well. This setting is only relevant if *SendEmailNotificationOfComment* is true or if you are using the support email functionality.

```
<add key=" SmtpServer" value="relay.yourmailserver.com" />
```

9. **EmailCredentialsUsername**. This is the optional username for your email credentials. On some systems, credentials are required in order for the mail server listed in *SmtpServer* to allow emails from your application.

```
<add key=" EmailCredentialsUsername" value="youremail@yourserver.com"
/>
```

10. **EmailCredentialsPassword**. This is the optional password for your email credentials. On some systems, credentials are required in order for the mail server listed in *SmtpServer* to allow emails from your application.

```
<add key=" EmailCredentialsPassword" value="xyz" />
```

11. **EmailCredentialsDomain**. This is the optional domain for computer name that verifies the other email credentials. It is often left blank. See http://msdn.microsoft.com/en-us/library/system.net.networkcredential.domain.aspx for more details.

```
<add key=" EmailCredentialsDomain" value="" />
```

12. **Label_UserName**. This is the label used to denote the “User” in emails. It is editable here for localization purposes.

```
<add key="Label_UserName" value="User:"/>
```

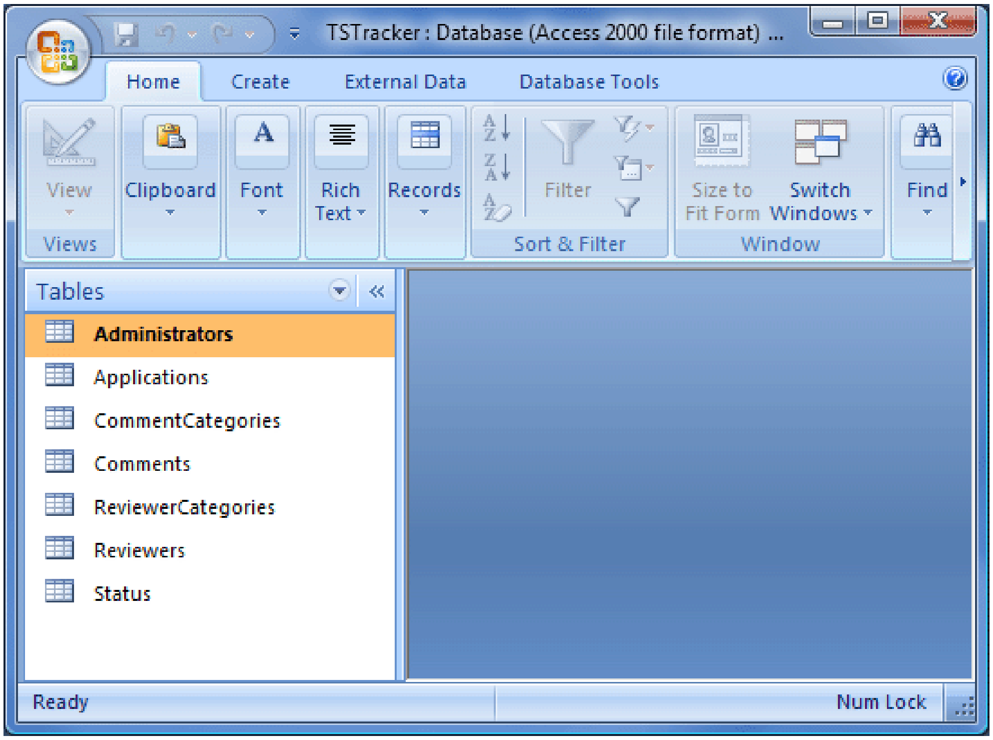
13. **Label_LessonName**. This is the label used to denote the “Lesson Name” in emails. It is editable here for localization purposes.

```
<add key="Label_LessonName" value="Lesson Name:"/>
```

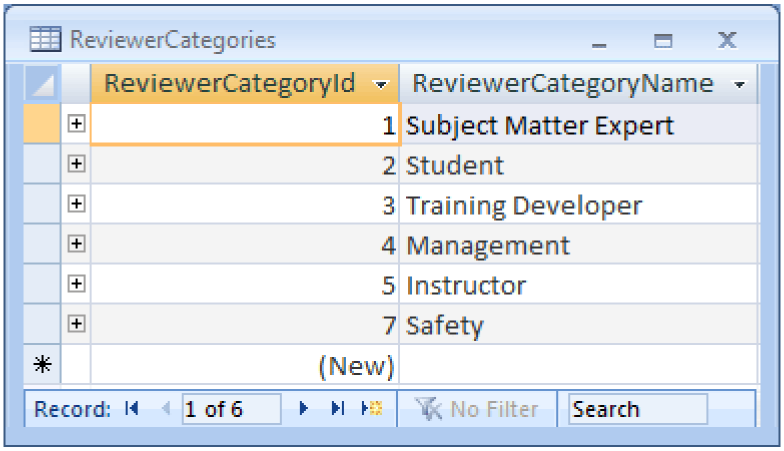
Editing the Tracking Database

As mentioned earlier in this document, all comments, reviewers, administrators, and status changes are stored in the *TSTracker.mdb* file located in the \App_Data directory. An Access database is used rather than a more powerful database like SQL Server since large numbers of simultaneous are not anticipated and many target end-user organizations may not have a SQL Server license. However, the web service may be edited to use other databases if desired. The tables within the database are shown to the right.

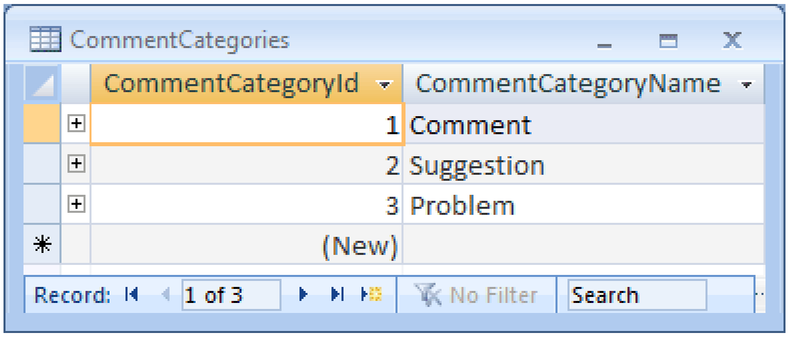
The *Administrators*, *Applications*, *Comments*, and *Reviewers* tables are intended to be populated and edited by the *Comments* and *Training Studio Tracker Administrator* applications. However, you may open the database directly to eliminate unneeded or obsolete information in them if desired. Note that you will typically need to download the database to your local machine before opening it in Access.



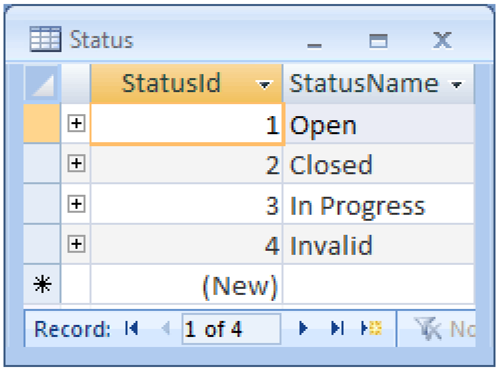
The *CommentCategories*, *ReviewerCategories*, and *Status* tables have no interface in other applications. If you want to change these, you will need to use Access. The *Comments* and *Training Studio Tracker Administrator* applications read these tables for the categories and will hence be automatically updated. You may rename categories and status elements but do not delete them if they have associated comments as that could cause errors. The layout of these tables is shown below and to the right.



ReviewerCategoryId	ReviewerCategoryName
1	Subject Matter Expert
2	Student
3	Training Developer
4	Management
5	Instructor
7	Safety
(New)	



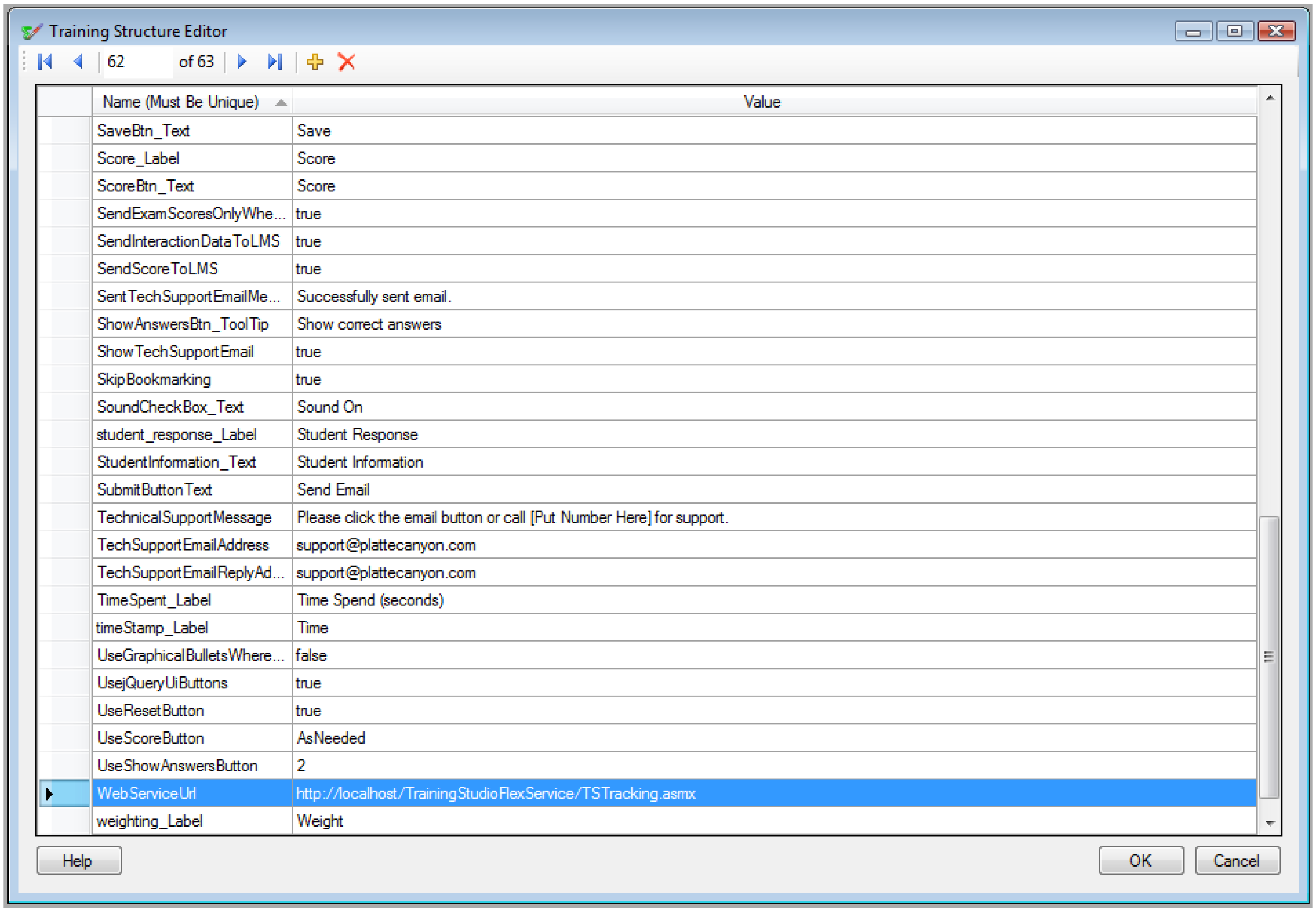
CommentCategoryId	CommentCategoryName
1	Comment
2	Suggestion
3	Problem
(New)	



StatusId	StatusName
1	Open
2	Closed
3	In Progress
4	Invalid
(New)	

Editing *Comments* Settings

The *Comments* functionality (and the rest of Training Studio) is written in HTML, JavaScript, and jQuery. Before deploying lessons with *Comments* functionality, you will need to change the *WebServiceUrl* in the Training Structure Editor of the *Training Studio Content Editor* as shown below.



Name (Must Be Unique)	Value
SaveBtn_Text	Save
Score_Label	Score
ScoreBtn_Text	Score
SendExamScoresOnlyWhe...	true
SendInteractionDataToLMS	true
SendScoreToLMS	true
SentTechSupportEmailMe...	Successfully sent email.
ShowAnswersBtn_ToolTip	Show correct answers
ShowTechSupportEmail	true
SkipBookmarking	true
SoundCheckBox_Text	Sound On
student_response_Label	Student Response
StudentInformation_Text	Student Information
SubmitButtonText	Send Email
TechnicalSupportMessage	Please click the email button or call [Put Number Here] for support.
TechSupportEmailAddress	support@plattecanyon.com
TechSupportEmailReplyAd...	support@plattecanyon.com
TimeSpent_Label	Time Spend (seconds)
timeStamp_Label	Time
UseGraphicalBulletsWhere...	false
UsejQueryUiButtons	true
UseResetButton	true
UseScoreButton	AsNeeded
UseShowAnswersButton	2
WebServiceUrl	http://localhost/TrainingStudioFlexService/TSTracking.asmx
weighting_Label	Weight

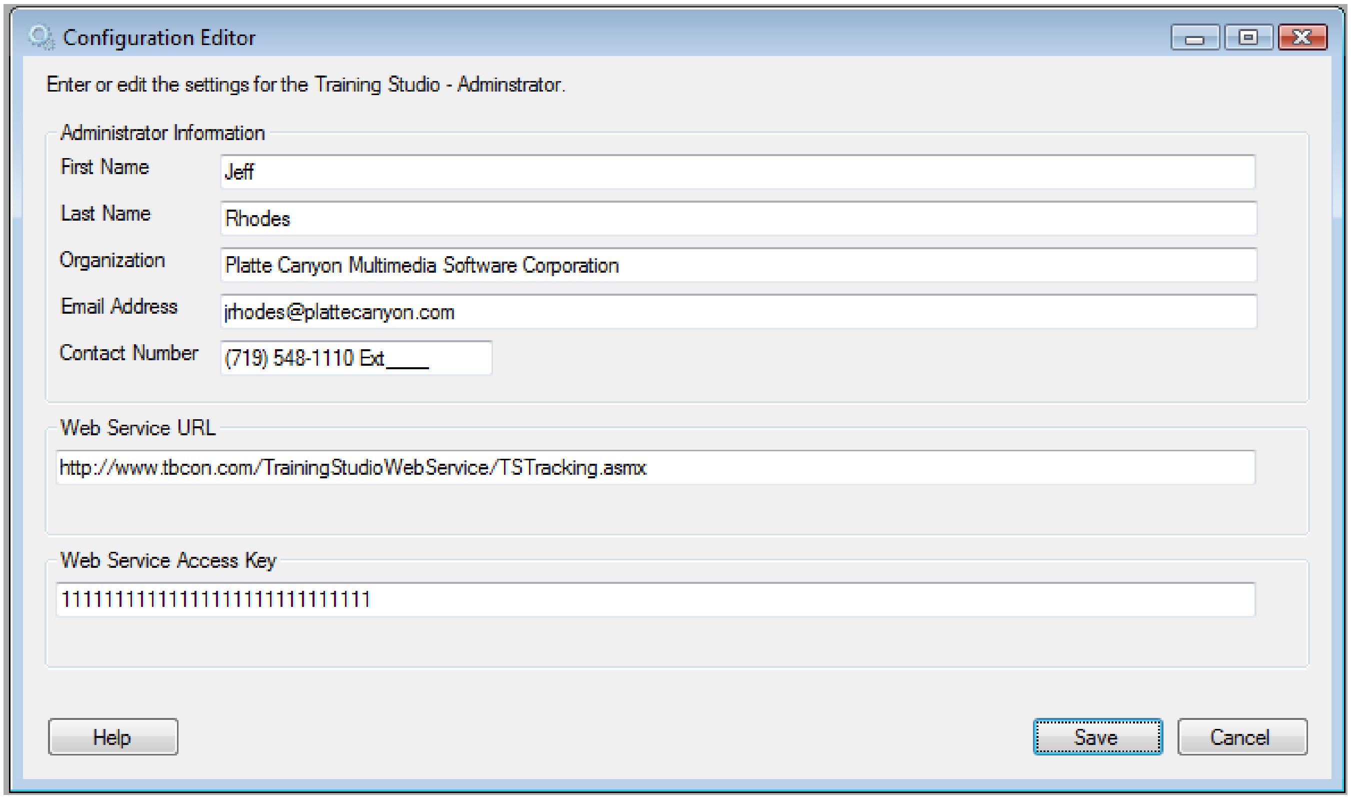
The updated structure then needs to be writing to the *TrainingStructure.xml* file and linked to by the *Training Studio Publisher*.

If desired, you may also change the *reviewKey* variable in TrainingStudioSupport.js within your Training Studio Base project to match the *Web.config* file as described above.

If you want to turn off the ability to send comments, set the *reviewOn* variable in TrainingStudioSupport.js to false. Otherwise, users can initiate comments by pressing Ctrl + Shift + M when the lesson has the focus.

Configuring the *Training Studio Tracker Administrator* Application

The URL to the web service as well as the *AdminTrackingKey* is needed by users of the *Training Studio Tracker Administrator* application in order for it to communicate with the web service and update comments. The *Training Studio Tracker Administrator* Configuration screen is shown below:



Configuration Editor

Enter or edit the settings for the Training Studio - Adminstrator.

Administrator Information

First Name: Jeff

Last Name: Rhodes

Organization: Platte Canyon Multimedia Software Corporation

Email Address: jrhodes@plattecanyon.com

Contact Number: (719) 548-1110 Ext____

Web Service URL

http://www.tbcon.com/TrainingStudioWebService/TSTracking.asmx

Web Service Access Key

1111111111111111111111111111

The user enters his or her contact information as well as the Web Service URL and Web Service Access Key (which must match the *AdminTrackingKey* defined in the *Web.config* file). Once the user saves this information, it is restored each time he or she opens the *Training Studio Tracker Administrator* application.