

ADOBE PDF TEST TOOLKIT

APTT User Guide **Automating PDF Testing for the Enterprise**

Installation

Testing User Guide

FAQ and Examples

Acrobat® Family of Products 9.2
Modification date: 10/7/09



Adobe® PDF Test Toolkit User Guide.

If this guide is distributed with software that includes an end user agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe, Acrobat®, Reader®, and the Adobe logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Windows®, Windows NT®, and Windows XP® are registered trademarks of Microsoft® Corporation registered in the United States and/or other countries. Mac® and Macintosh® are registered trademarks of Apple Computer®, Inc. in the United States and other countries. All other trademarks are the property of their respective owners.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA. Notice to U.S. Government End Users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Contents

1	Installation and Configuration	4
1.1	What is the Adobe PDF Test Toolkit?	4
1.2	System requirements	4
1.3	Installation	5
1.3.1	Install the PDF viewing client.....	5
1.3.2	Install the Adobe PDF Test Toolkit	5
1.3.3	Install components to support Flex testing	6
1.3.3.1	Install Flex Builder.....	6
1.3.3.2	Install the Flex plug-in for Quick Test Pro	6
1.3.4	Verify the server (LiveCycle ES) software	8
1.3.5	Install Quick Test Professional.....	8
1.3.6	Vista and QTP	10
1.4	Uninstalling the Adobe PDF Test Toolkit.....	11
1.5	Registry changes	11
2	Guidelines for Creating Tests	14
2.1	Guidelines for successful tests.....	14
2.2	Supported form elements	14
2.3	Creating a new test script.....	15
2.3.1	LiveCycle testing	15
2.3.2	Acrobat and Adobe Reader testing	16
2.3.3	Browser viewer testing	17
2.4	Testing events	17
2.5	Checkpoint properties.....	20
2.6	Learning objects in the object repository.....	21
2.7	Miscellaneous FAQs and operational details	22
2.8	Errors.....	23
3	Example QTP Tests	24
3.1	XML form in a browser.....	24
3.2	Accessing unexposed properties.....	25
4	Index.....	26

1 Installation and Configuration

1.1 What is the Adobe PDF Test Toolkit?

The Adobe PDF Test Toolkit (APTT) is a Windows-based set of tools for the Acrobat family of products version 9.2 and later. It is designed to automate PDF forms workflow testing, including the following:

- Reader and Acrobat running both stand-alone and within a browser.
- LiveCycle ES and SAP.
- XML dynamic forms (XFA 3.0 for LiveCycle Designer; XFA 2.9 when workflows include the LC ES server).
- Acroforms.
- Documents that have been “reader enabled” so that Adobe Reader users can perform actions that would otherwise be prohibited.

This early release is designed for integration with HP Quick Test Professional (QTP). While this product is “beta”, we believe it is stable enough to use to test production systems that utilize Adobe Reader and PDF Forms. Your interest, feedback, and comments will determine future development of the toolkit.”

Note: QTP is a functional user interface test automation tool from Hewlett Packard that enables user interface-based testing of software applications in complex workflows.

1.2 System requirements

In addition to the software and hardware requirements listed below, the person installing the APTT plugins on the client must have permissions to modify files and registry entries.

- All the system requirements defined by Acrobat or Adobe Reader 9.2.
- While APTT is only provided in English, it may be run in all multilingual and localized environments.

Table 1 Software and hardware requirements

Category	Requirement	Notes
Client software		
	Acrobat 9.2 or Adobe Reader 9.2	Uninstall previous versions.
	Flex Builder 3.0.2	Optional: Only needed if testing against LiveCycle UI which uses Flex. 3.0 users should install the 3.0.2 update.
	Flex QTP Plugin	Included with Flex Builder
	HP Quick Test Pro 9.5 with updates or version 10	
Server software		When testing includes LiveCycle ES workflows
	LiveCycle Server 8.2 or later	

Table 1 Software and hardware requirements

Category	Requirement	Notes
Operating system		
	Windows XP SP3	Requires IE 6 or later
	Windows Vista sp1	Requires IE 7 or later
Browser		
	IE 6 or 7	For Windows XP
	IE 7	For Windows Vista
Hardware	None	APTT supports the requirements specified by the Acrobat family of products and HP Quick Test Pro.

1.3 Installation

Environment configuration involves the following:

- [Install the PDF viewing client](#)
- [Install the Adobe PDF Test Toolkit](#)
- [Install components to support Flex testing](#)

Tip: When testing scenarios involve testing LiveCycle ES user interface components, install the requisite version of Flex Builder and the related plugins. If LC ES components are not tested, skip those steps.

- [Verify the server \(LiveCycle ES\) software](#)
- [Install Quick Test Professional](#)
- [Vista and QTP](#)

1.3.1 Install the PDF viewing client

Install the appropriate PDF viewing software:

1. Uninstall any versions of Acrobat or Adobe Reader older than 9.2.
2. Install any 9.2 or later Adobe Reader or Acrobat product. Only 9.2 and later is supported and the directories, registry entries, and so on must exist prior to installing APTT.

1.3.2 Install the Adobe PDF Test Toolkit

Before installing APTT, make sure you have the requisite versions of Acrobat and LiveCycle and that all of the other system requirements are met.

To install APTT components:

1. Go to <https://prerelease.adobe.com/>, and log in.
2. Navigate to the PDF Test Toolkit area.

3. Download PDFTestToolkit_<version>.zip.
4. Extract the file to any location.
5. Go to the Toolkit's plugins directory <Toolkit install location>\plugins\.
6. Copy the two plugins the install registration file to your PDF client's plugin directory: <PDF viewer install location>\Adobe\<product name><version>\<product>\plug_ins\
 - AcroQTP.exe
 - Automation.api
 - regPDFTestToolkit.bat
7. Double click the regPDFTestToolkit.bat file to register the plugins.
8. Press **Enter** when the command prompt verifies successful installation.

1.3.3 Install components to support Flex testing

Most testing includes testing XML forms with the LiveCycle ES Flex-based web interface. However, if it doesn't, do not install the Flex 3 plugin for QTP, and skip to ["Install Quick Test Professional" on page 8](#).

1.3.3.1 Install Flex Builder

1. Verify you have installed Flex Builder 3.0.2.

Tip: You need both Flex Builder 3.0 and the 3.0.2 update which is downloaded separately.

If you don't have the requisite software, see the following:

- **Flex Builder 3.0:** <http://www.adobe.com/products/flex/>
- **Flex Builder 3.0.2 update:** http://www.adobe.com/support/flex/downloads_updaters.html#flex3

2. Install and configure Flex Builder according to the Flex Builder documentation.

Tip: You do not need the Flex Automated Testing plugin or any of its configuration steps to use the Adobe PDF Testing Toolkit.

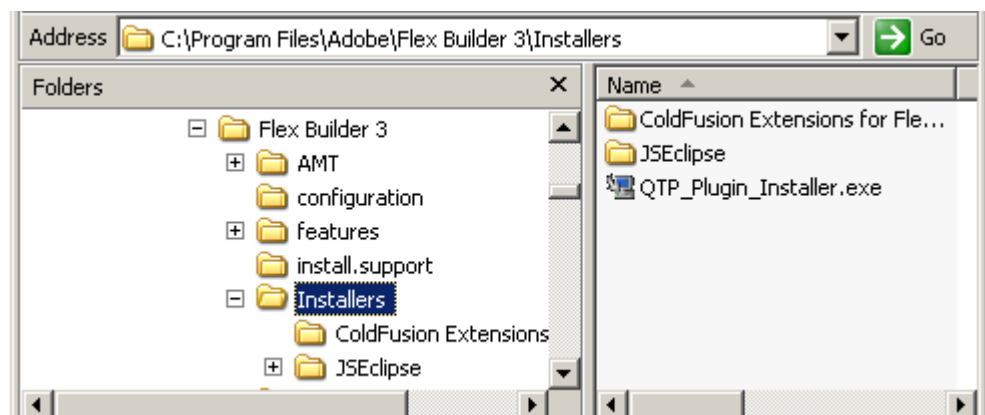
1.3.3.2 Install the Flex plug-in for Quick Test Pro

install the Adobe Flex 3 Plug-in for Mercury Quick Test Pro. This plugin enables testing of LiveCycle's Flex-based user interface components.

There are two ways to get the plugin:

- **Flex Builder 3.0:** If you already have Flex Builder 3.x installed, you probably already have the installer in the Installer directory ([Figure 1](#)).
- **Web download:** Download the installer from http://www.adobe.com/support/documentation/en/flex/3/releasenotes_flex3_fb.html#install_automated_testing.

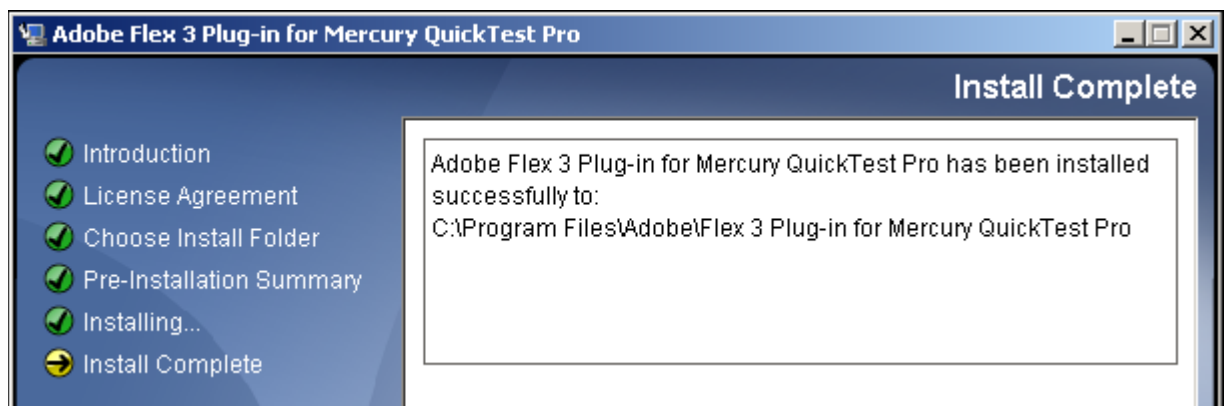
Figure 1 Flex plugin directory



To install the Flex plugin:

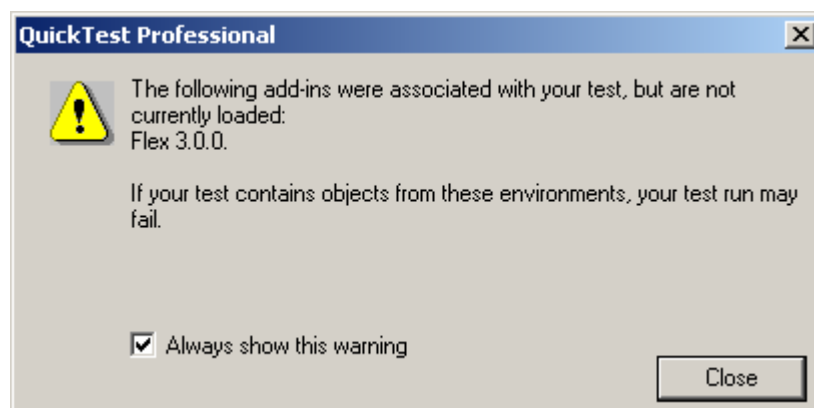
1. Run the installer.
2. Go through all the screens and choose **Finish**.
3. Restart your machine.

Figure 2 Flex QTP plugin installer



Note: If a test requires this component and it is not installed, a warning may appear.

Figure 3 QTP: Missing Flex components warning



1.3.4 Verify the server (LiveCycle ES) software

APTT supports LiveCycle ES 8.2. While other versions may work, they are not officially supported at this time.

The Flex user interface is built using the Flex framework, and when compiled with automation libraries (or when the runtime loader loads it) that alone will make the QTP plugin work. The server does not do anything in particular to support automated testing, and no special configuration is required.

Test URLs

Note that for testing, record and run properties must occur against a test URL for which the automation bits are turned on in the server's automation SWFs. These bits are on by default. For example, if LiveCycle's main URL is:

```
http://<server name>:<server port>/<installation path>/Main.html
```

Then the URL to use for testing would be:

```
http://<server name>:<server port>/<installation path>/Main-auto.html
```

1.3.5 Install Quick Test Professional

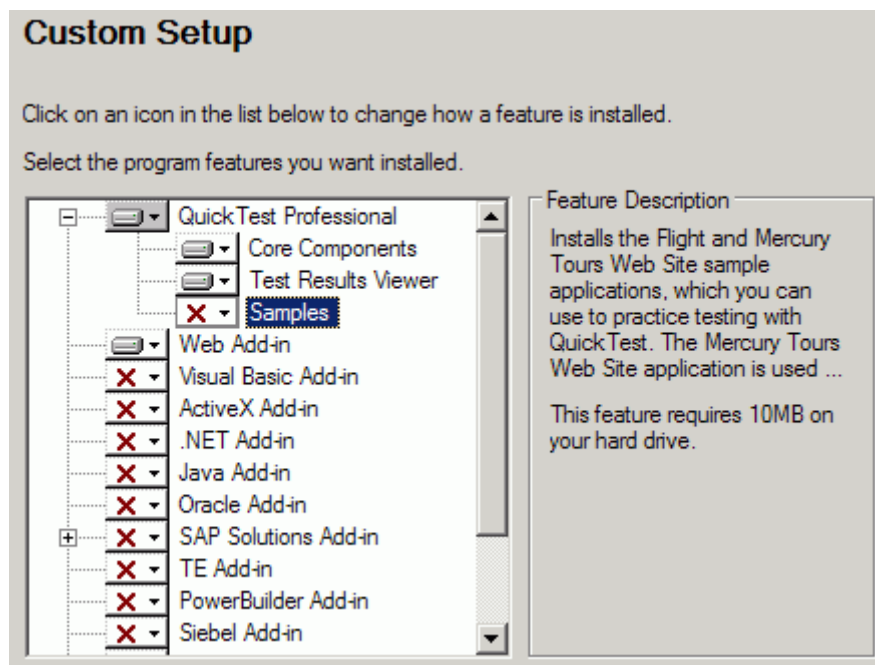
If you don't already have Quick Test Professional 9.5 installed, install it now.

1. Run the installer.

Note: Running the installer automatically overwrites old plugin files. You do not need to delete them.

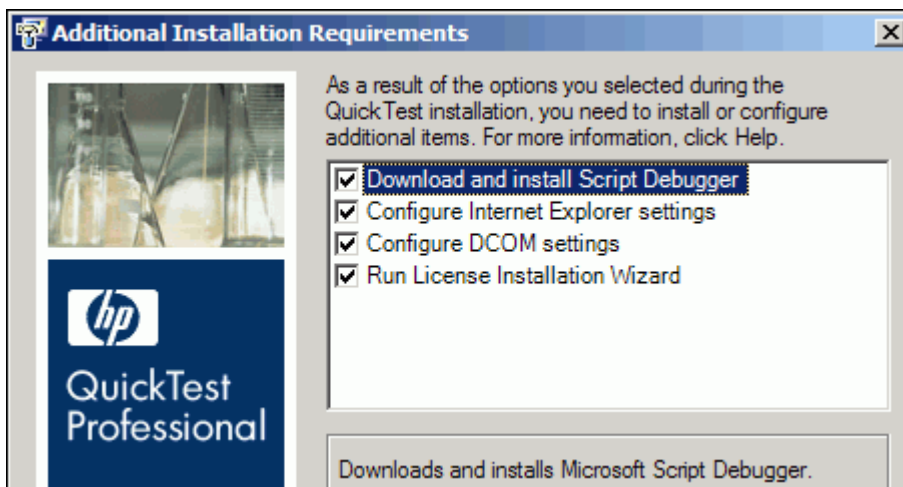
2. Install any required components the installer asks you to; for example, Microsoft Visual C++ Run-time Components.
3. Choose **Next** until you get to the Setup Type screen.
4. You can accept the basic installation; however, some components aren't needed. If you'd like to only install the components you need, do the following:
 1. Choose **Custom**.
 2. The following components are not required as part of the PDF Test Toolkit. Deselect them if you don't need them for other testing and would like a minimal install:
 - ActiveX Add-in
 - Visual Basic Add-in
 - Samples (under Quick Test Professional)
 3. Choose **Next** and continue the installation normally.

Figure 4 QTP: Custom installation



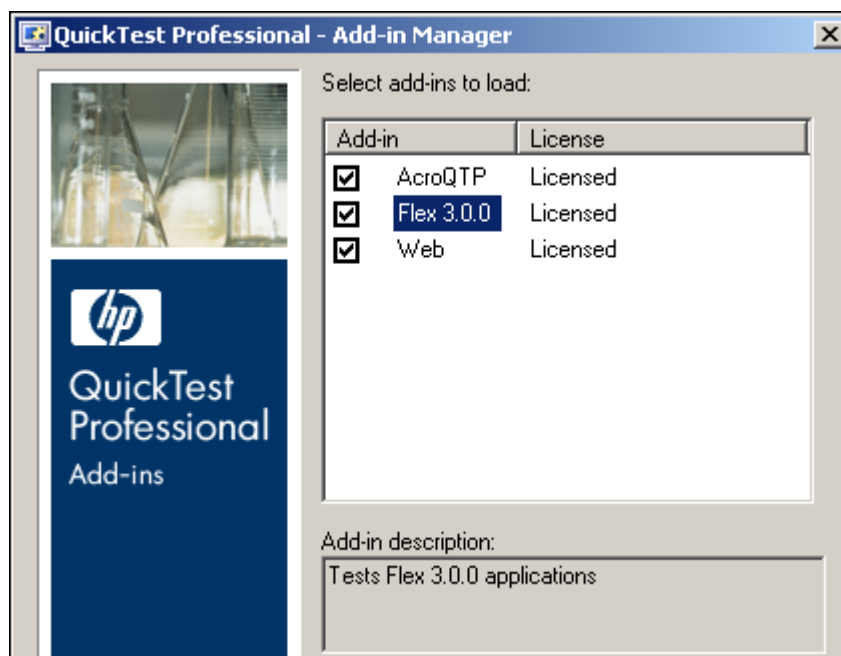
5. Choose **Finish** when done.
6. If an Additional Installation Requirements screen appears, choose any items or choose **Cancel**. APTT does not require any special configuration with respect to these items.

Figure 5 QTP: Additional installation components



7. Launch QTP.
8. If the Add In Manager appears, verify the needed add ins are checked:
 - **AcroQTP**: Required.
 - **Flex 3.0.0**: Required when testing Flex-based LiveCycle ES web user interface.
 - **Web**: Required when testing Flex-based LiveCycle ES web user interface.
9. Choose **OK**.

Figure 6 QTP: Add-In Manager



10. Install QTP updates. QTP should display a dialog on startup if you don't have the latest updates.

Tip: It may be useful to verify your system is set up correctly. To do so, load and run on of the APTT's test files included examples as described in ["XML form in a browser" on page 24](#).

1.3.6 Vista and QTP

When testing on a machine running Vista, perform the following steps:

Configure IE

1. Open Internet Explorer.
2. Choose **Tools > Internet Options > Advanced**.
3. Check **Enable 3rd party extensions**.

Clear the User Account Control (UAC)

1. Log in as an administrator.
2. Choose **Control Panel > User Accounts > Change Security Settings**.
3. Uncheck **User Account Control (UAC) to help protect your computer**.
4. Choose **OK**.
5. Restart the machine.

1.4 Uninstalling the Adobe PDF Test Toolkit

To uninstall APTT components:

1. Go to the Toolkit's plugins directory <\$APTT install root>\plugins\.
2. Double click the unregPDFTestToolkit.bat file to unregister the plugins.
3. Delete all the APTT files.

1.5 Registry changes

The registry is automatically modified when the plugins are registered. The following is provided for informational purposes only during the beta releases of APTT.

Since there is no uninstaller, removal of the plugins leaves the entries shown in [Example 1.1](#).

Example 1.1 Registry entries

```
[HKEY_CLASSES_ROOT\AcroQTP.Proxy]
@="AcroQTP Proxy Class"

[HKEY_CLASSES_ROOT\AcroQTP.Proxy\CLSID]
@="{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}"

[HKEY_CLASSES_ROOT\AcroQTP.Proxy\CurVer]
@="AcroQTP.Proxy.1"

[HKEY_CLASSES_ROOT\AcroQTP.Proxy.1]
@="AcroQTP Proxy Class"

[HKEY_CLASSES_ROOT\AcroQTP.Proxy.1\CLSID]
@="{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}"

[HKEY_CLASSES_ROOT\AppID\{78B32EBB-4F99-4ee7-A2DE-3655652719B2}]
@="AcroQTP.Proxy"

[HKEY_CLASSES_ROOT\AppID\AcroQTP.exe]
"AppID"="{78B32EBB-4F99-4ee7-A2DE-3655652719B2}"

[HKEY_CLASSES_ROOT\CLSID\{27E18E9E-9C34-4ce2-B19C-3000B8D5E2B2}]
@="AcroQTP Plugin Class"

[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}]
@="AcroQTP Proxy Class"
"AppID"="{78B32EBB-4F99-4ee7-A2DE-3655652719B2}"

[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}\LocalServer32]
@="\"C:\Program Files\Adobe\Acrobat 9.2\Acrobat\plug_ins\Test_Tools\AcroQTP.exe\"

[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}\ProgID]
@="AcroQTP.Proxy.1"
```

```
[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}\Programmable]

[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}\TypeLib]
@="{2AABF103-0E9C-4f15-9F4E-3D344DFB9A30}"

[HKEY_CLASSES_ROOT\CLSID\{BF95D3DB-AC03-46f1-97F1-C4B91F1A8F20}\
VersionIndependentProgID]
@="AcroQTP.Proxy"

[HKEY_CLASSES_ROOT\Interface\{82718C32-93CD-4ECC-BD6F-0F5AF70C4E89}]
@="IAcroQTPProxy"

[HKEY_CLASSES_ROOT\Interface\{82718C32-93CD-4ECC-BD6F-0F5AF70C4E89}\
ProxyStubClsid]
@="{00020424-0000-0000-C000-000000000046}"

[HKEY_CLASSES_ROOT\Interface\{82718C32-93CD-4ECC-BD6F-0F5AF70C4E89}\
ProxyStubClsid32]
@="{00020424-0000-0000-C000-000000000046}"

[HKEY_CLASSES_ROOT\Interface\{82718C32-93CD-4ECC-BD6F-0F5AF70C4E89}\TypeLib]
@="{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}"
"Version"="1.0"

[HKEY_CLASSES_ROOT\Interface\{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}]
@="IAcroQTPPlugin"

[HKEY_CLASSES_ROOT\Interface\{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}\
NumMethods]
@="11"

[HKEY_CLASSES_ROOT\Interface\{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}\
ProxyStubClsid]
@="{00020424-0000-0000-C000-000000000046}"

[HKEY_CLASSES_ROOT\Interface\{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}\
ProxyStubClsid32]
@="{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}"

[HKEY_CLASSES_ROOT\Interface\{A38CF77B-FBA8-43B1-B38B-2A7C4486337B}\TypeLib]
@="{3AABCA85-0B01-48A6-86C0-9DECC1C50854}"
"Version"="1.0"

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}]

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}\1.0]
@="AcroQTP Proxy 1.0 Type Library"

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}\1.0\0]

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}\1.0\0\win32]
@="C:\Program Files\Adobe\Acrobat 9.2\Acrobat\plug_ins\Test_Tools\
AcroQTP.exe"

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}\1.0\FLAGS]
@="0"

[HKEY_CLASSES_ROOT\TypeLib\{2AABF103-0E9C-4F15-9F4E-3D344DFB9A30}\1.0\HELPDIR]
@=""
```

```
[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}]

[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}\1.0]
@="AcroQTP Plugin 1.0 Type Library"

[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}\1.0\0]

[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}\1.0\0\win32]
@="C:\\Program Files\\Adobe\\Acrobat 9.2\\Acrobat\\plug_ins\\Test_Tools\\Automation.api"

[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}\1.0\FLAGS]
@="0"

[HKEY_CLASSES_ROOT\TypeLib\{3AABCA85-0B01-48A6-86C0-9DECC1C50854}\1.0\HELPDIR]
@=""
```

2

Guidelines for Creating Tests

Testing methodologies will vary slightly with the nature of the form being tested and testing environment. For example, prior to testing, the developer should know the following:

- Is LiveCycle ES part of the workflow? Testing LiveCycle requires Flex components and a special test URL.
- What is the nature of forms in the workflow? There are several considerations:
 - Forms created with LiveCycle Designer ES are XML based and can have dynamic elements.
 - Forms created with Acrobat (Acroforms) are PDF language based.
 - Is the form “reader enabled” and given usage rights so that Adobe Reader users can perform actions that would otherwise be prohibited.
- Will forms be exercised in a standalone PDF viewer (Adobe Reader or Acrobat), or will they be exercised while viewed within a browser shell?

2.1 Guidelines for successful tests

The following guidelines will help you successfully record and playback tests:

- Run tests on the machine on which they were recorded.
- Record and run tests with only one Adobe client running.
- Record tests with only one PDF file open.

2.2 Supported form elements

The form elements supported by this release include those in [Table 4](#). Note the following:

- Forms created with LiveCycle Designer ES are XML based and can have dynamic elements.
- Forms created with Acrobat (Acroforms) are PDF language based.

Tip: LiveCycle forms are XML based. forms. Thus, forms created by LiveCycle Designer are often referred to as XML forms. XFA is the name of the XML LiveCycle user.

Table 4 Supported form elements

Element	Acroform	XML (XFA)	Description
PDF Document	Yes	Yes	PDF Document
AcroField	N/A	N/A	Not visible within QTP
Button	Yes	Yes	Button
CheckBox	Yes	Yes	Check Box

Table 4 Supported form elements

Element	Acroform	XML (XFA)	Description
ComboBox	Yes		Combo Box
ListBox	Yes	Yes	List Box
RadioButton	Yes	Yes	Radio Button
RadioButtonList	Yes	Yes	Radio Button group
Signature	Yes	Yes	Signature field
Text	Yes	Yes	Text field
XFAForm	N/A	Yes	XFA Form
Table	N/A	Yes	Table
TableHeader	N/A	Yes	Table Header
TableRow	N/A	Yes	Table Row

2.3 Creating a new test script

2.3.1 LiveCycle testing

When LiveCycle ES is part of the tested workflow, verify you have the requisite Flex components installed as described in [Chapter 1, "Installation and Configuration"](#).

Note that for testing, record and run properties must occur against a test URL for which the automation bits are turned on in the server's automation SWFs. These bits are on by default. When you start recording, set this URL. For example, if LiveCycle's main URL is:

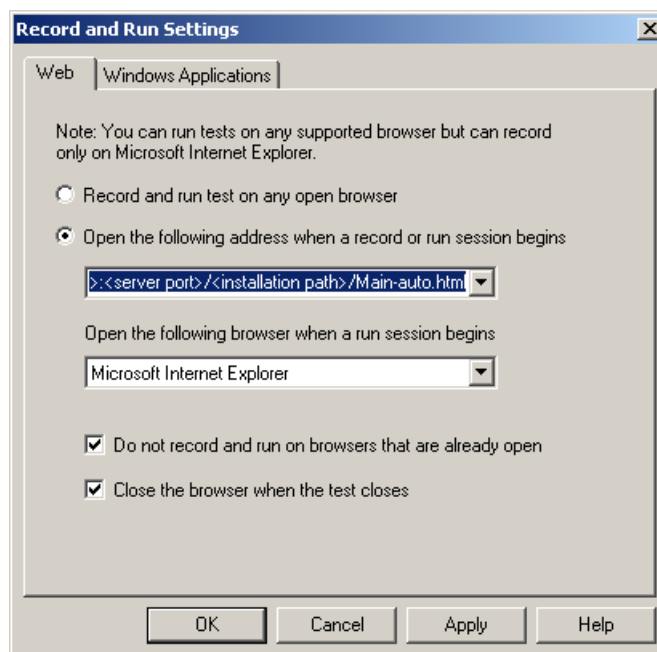
```
http://<server name>:<server port>/<installation path>/Main.html
```

Then the URL to use for testing would be:

```
http://<server name>:<server port>/<installation path>/Main-auto.html
```

Tip: Additional information pertinent to testing within the LiveCycle environment can be found at http://livedocs.adobe.com/flex/3/html/agents_1.html.

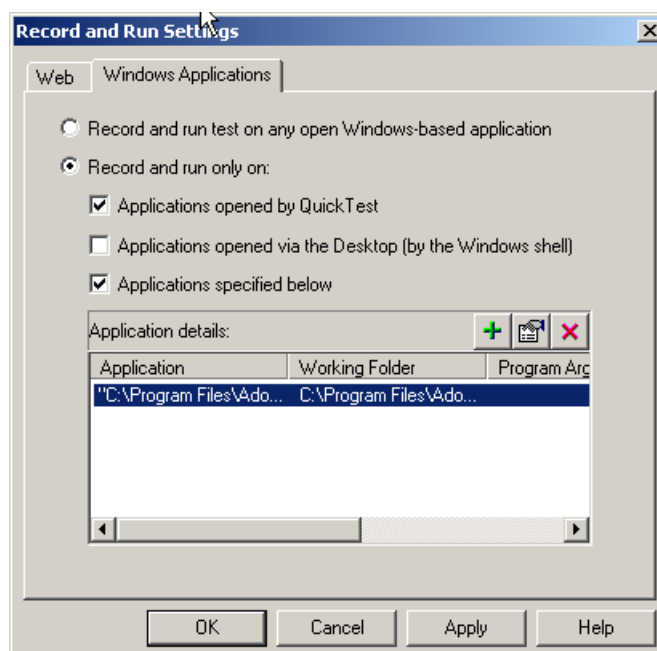
Figure 7 QTP configuration: Pointing to the LiveCycle automation URL



2.3.2 Acrobat and Adobe Reader testing

There are no special steps required, however, if you would like QTP to invoke the standalone application as part of the test (as opposed to having the document already open), configure the record and run settings so that QTP can open the application.

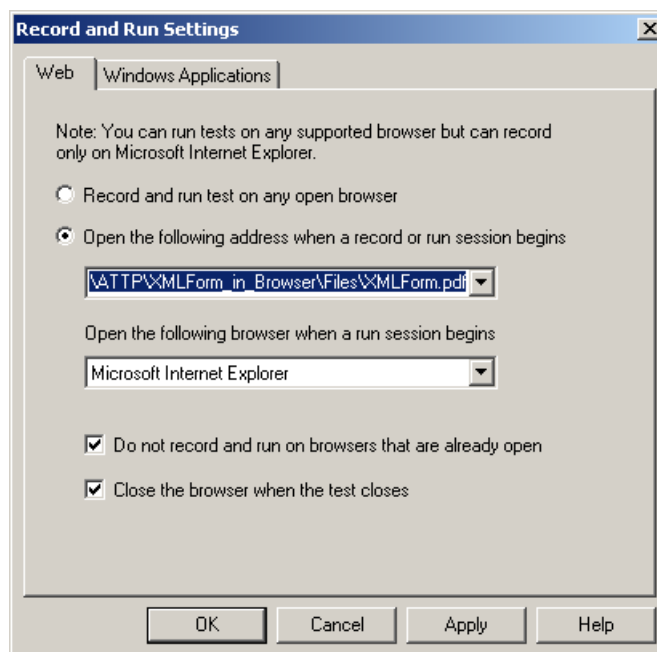
Figure 8 QTP configuration: Invoking the standalone PDF viewer



2.3.3 Browser viewer testing

There are no special steps required, however, if you would like QTP to open a PDF in the browser, (as opposed to having the document already open), configure the record and run settings so that QTP can use an URL to find the file.

Figure 9 QTP configuration: Invoking the standalone PDF viewer



2.4 Testing events

Capturing events and other operations are an essential part of forms testing. [Table 5](#) lists the supported operations.

Note: In QTP, the operations appear in the Operations column. APTT's supported operations appear in addition to the standard events provided by QTP.

Table 5 Operations provided and supported by the PDF Test Toolkit

Operation	Description
Click	Generate a mouse click on the center of the field.
Select	Select the form field.
SetValue	Set the value of the field to specified value.
Typetext	Generate keystroke events for the field.
ClearEvents	Clear the event history.
EnableEvents	Enable capturing the event history.

Table 5 Operations provided and supported by the PDF Test Toolkit

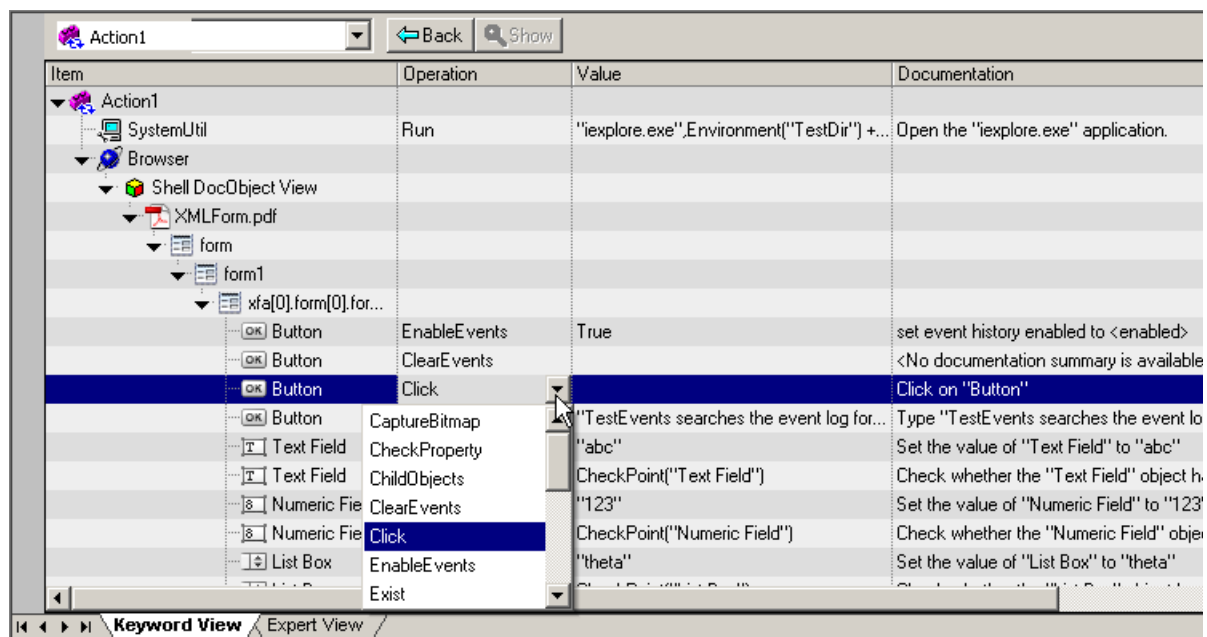
Operation	Description
GetEvent	Set the property 'Event' to value of the specified event. Events are identified by index starting with 0 to 'NumEvents' - 1.
TestEvents	Test the event log to see if it contains the specified string and sets two values: <ul style="list-style-type: none"> • TestEventsPassed: A boolean value indicating the success of the test. • Event: The value of the matching property.
SetTimeOut	A PDFDoc method used to set the 'TimeOut' property to the specified value.

To add an event test on a form field such as a button or text field, perform the following:

1. Perform a form field action such as a button click.
2. Enable the events on the field you want to test:
 1. Select the row that contains the form field for which you want to add an action. In this example, it is a button.
 2. Right click and copy that row (including the action item, operation, value and documentation).
3. Paste the clipboard to the same row 4 TIMES. QTP will paste the new rows right above the old row. Now configure the three new rows:
 1. In the first row's operation column, select the operation cell and choose **Enable Events** from the drop down list. In the value column, select **True**.
 2. In the second row's operation column, select the operation cell and choose **Clear Events** from the drop down list. This clears the events so no previous events remain in the event stream.
 3. In the third row's operation column, select the operation cell and choose **Click** from the drop down list.
 4. In the fourth row's operation column, select the operation cell and choose **TestEvents** from the drop down list.
 5. For TestEvents add a value to the event name followed by space followed by full somExpression for the field you are adding the event to. For example, "click: xfa[0].form[0].form1[0].#subform[0].Button1[0]".

Tip: Make sure the event name is all lower case and there is a space next to the colon.

Figure 10 Event testing: Selecting the event



4. Add a check point to make sure the event has occurred.

Tip: The PDF test file needs to be open before setting a checkpoint.

1. Select the row containing the TestEvents operation.
2. Choose **Insert > Checkpoint > Standard Checkpoint**.
3. From the list of property values displayed uncheck all except testEventsPassed.
4. Set the value 1.
5. Stop recording.
6. Run the test.
7. Review the results:
 1. In the Test Results window on the left hand side, select the **TestClickIteration**.
 2. Right click and choose **Expand all**.
 3. Look for a green check mark or a red X mark indicating a pass or fail respectively.
 4. For each red X, select that action and double click.
 5. Compare the provided value with the test result value.

2.5 Checkpoint properties

The following tables list the supported checkpoint properties.

Any XML dynamic form or Acroform props not explicitly listed can still be used via QTP. To do so, enter the property manually along with the desired value. For more information, see the following:

- **Acroforms:** http://www.adobe.com/devnet/acrobat/pdfs/js_api_reference.pdf
- **XML forms:** http://help.adobe.com/en_US/lifecycle/8.2/lcdesigner_scripting_reference.pdf
- **XFA specification:** http://partners.adobe.com/public/developer/xml/index_arch.html#xfa_28

Table 6 Generic properties not subject to user customization

Property	Description
afpModel	AcroQTP object model: not subject to user customization.
afpType	AcroQTP object type: not subject to user customization.
afpld	AcroQTP object ID: not subject to user customization.
ready	A PDFDoc property indicating the application is ready to perform the next operation: not subject to user customization.

Note: The following tables provide only a partial listing of available properties.

Table 7 Acroform field properties

Property	Description
hidden	Acrobat field identifier
name	The name of the form element
numFields	Acrobat document number of AcroForm fields
noontimes	Acrobat field number of list items
page	Acrobat field page number
rent	Acrobat field rectangle
read-only	Acrobat field access
type	Acrobat field type
userName	Acrobat field name
value As String	Acrobat field string value

[Table 8](#) lists properties APTT provides to test various aspects of XFA objects.

Table 8 PDF Test Toolkit properties

Property	Description
timeout	Maximum milliseconds to wait for run command to complete.
event	The event resulting from a GetEvent or TestEvents operation.
events Enabled	A boolean value indicating the event logging state.

Table 8 PDF Test Toolkit properties

Property	Description
numEvents	Number of events in the event log - the maximum size of the log is 32 entries.
testEventsPassed	A boolean value indicating the success of the last TestEvents operation.

Table 9 XML form field and QTP properties

Property	Description
timeout	Maximum milliseconds to wait for run command to complete.
event	The event resulting from a GetEvent or TestEvents operation.
eventsEnabled	A boolean value indicating the event logging state.
numEvents	Number of events in the event log - the maximum size of the log is 32 entries.
testEventsPassed	A boolean value indicating the success of the last TestEvents operation.
access	field access
borderColor	field border color
borderWidth	field border width
highlight	field highlight
presence	field visibility
revalue	field raw value
rotate	field rotation
size	field size
somExpression	field identifier
x	field x position
y	field y position
h	field height
w	field width
Value	The field value. This value is an interpreted field value to simplify test generation. This value may be passed to the SetValue operation to set the field to the appropriate value.

2.6 Learning objects in the object repository

1. Open the PDF file containing the objects to add to the repository.
2. Choose the Add Objects icon (the icon with + sign) or the menu **Object > Add Objects to Local**.
3. QTP minimizes so that you can choose the form or the form field.
4. To learn only one object, click on the requisite form field; otherwise, click anywhere outside of a field.
5. Choose **Define Object Filter**, and make the appropriate selection.

6. Choose **OK**.

The form fields are now added to the repository.

2.7 Miscellaneous FAQs and operational details

Does APTT work with QTP version 10?

Yes. In fact, much testing has been done with version 10.

Using 10 has a benefit in that it does not record AVPageView statement for every click within a forms opened in Reader.

While recording in Adobe Reader, QTP 9.5 records an AVPageView statement for every click

The issue does not appear in QTP 10. Delete the AVPageView statements before running your tests. Users who do not record tests but rather manually draft them will not encounter this minor issue.

Toggling between forms

While recording a test that toggles between two forms, user has to first click on the title bar of the form for which he/she needs to record the actions. Otherwise the tool is not aware that the focus has shifted to a new form for which it has to record the actions.

An Acrobat recorded test fails when run on Adobe Reader

A test recorded on one client may or may not run successfully in another client. It is best to record tests for a particular client.

How do I identify specific actions associated with fields?

QTP truncates tree element names, so to figure out which field corresponds to which actions, go the checkpoint properties and look at the `afpid` value. Total name length is limited to 30 characters.

Why does QTP sometimes disappear during recording?

QTP minimizes and the icon disappears from the task bar while in a record session when the user inserts a check point or output value or uses the Object Repository feature to learn the objects. QTP has not crashed; it is still open and recording.

Can recorded tests be run on other machines?

Moving a client's position on the screen during recording does not impair a test; however, when a test is run on another machine, the test and recording is likely to fail because QTP might not be able to track the client. Differences in screen resolution may also have a negative impact.

Can a test be recorded at different points in time?

Possibly. However, some users encounter problems when recording tests on a LiveCycle server that have been partially recorded at different times. To ensure a clean and operational test. Record your test from beginning to end at one time.

Can more than one client or PDF be open during recording?

Recording is most reliable when occurring on one machine, with only one client running (Acrobat or Adobe Reader), and only one PDF open. QTP may record incorrectly if the testing environment contains too many variables.

Can I record digitally signing documents?

Yes. However, you must already have a digital ID and a valid certificate available. QTP is not able to record all the steps or creating a digital ID on the fly.

2.8 Errors

Failed to find object: Failed to find Acrobat

An unsupported version of Acrobat is being used or both Acrobat and Reader are installed.

3 Example QTP Tests

Experienced QTP users probably don't need help on setting up their environment or running the examples. However, if your first exposure to QTP is the result of using the Adobe PDF Test Toolkit, do the following:

- Install the toolkit in some convenient location.
- Create a directory where you'll deposit the results of your example tests.
- Follow all the steps described in [Chapter 1, "Installation and Configuration"](#).

The current release includes the following examples in <Toolkit installation directory>\test_files\:

- [XML form in a browser](#)
- [Accessing unexposed properties](#)

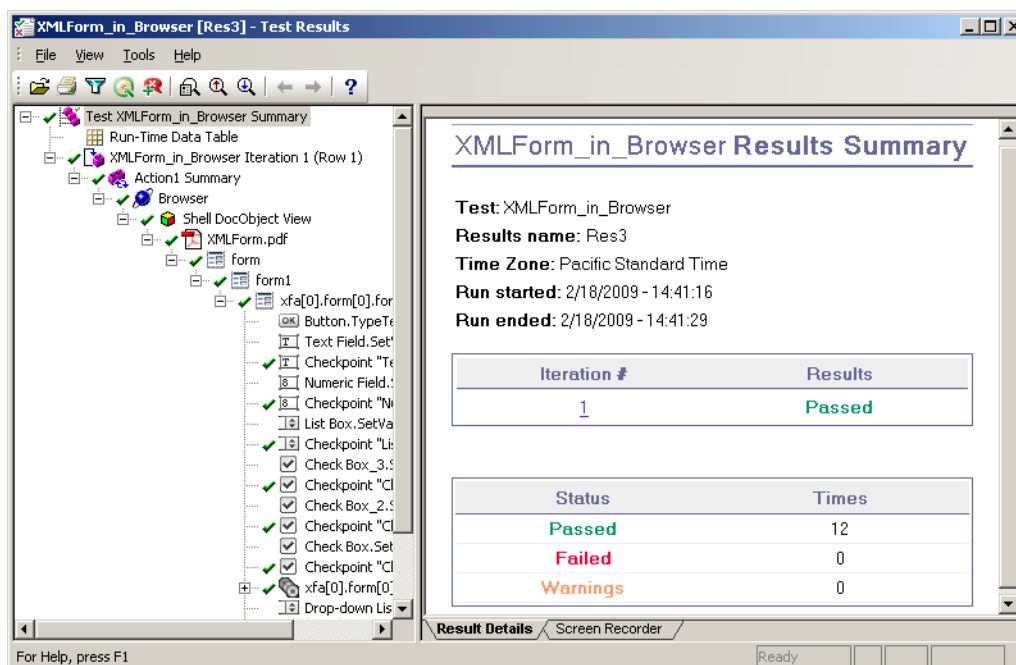
3.1 XML form in a browser

This simple test requires no special configuration. To run it:

1. Open QTP.
2. Choose **File > Open > Test**.
3. Browse to <Toolkit installation directory>\test_files\XMLForm_in_Browser.
4. Highlight XMLForm_in_Browser.
5. Choose **Open**.
6. Choose **Run**.

The test should run successfully with no failures.

Figure 11 Example QTP test: XML form in browser



3.2 Accessing unexposed properties

Note: The example files for this test reside in <Toolkit installation directory>\test_files\CheckingUnexposedPropVals.

Not every property is exposed in the QTP's check point's standard properties dialog. Some properties are several layers down in a hierarchy and must be manually configured. The following example shows how to test for the border color of a TextField.

1. Copy the step containing an operation; for example, a "select" of the TextField1 for which you are checking the property border color twice.
2. Edit the first copy of this step: Highlight the Select operation of this step, and change it to GetROProperty from the drop down menu.
3. In the value column, enter the SOM to the property you are testing; in this case it will be xfa.form.form1.TextField1.border.edge.color.value.
4. Edit the second copy of the step you copied in step 1 above: Highlight the Select Operation and change it to CheckProperty from the drop down menu.
5. In the Value column, enter the SOM expression xfa.form.form1.TextField1.border.edge.color.value.
6. In the next sub column of the Value column of the step, enter 0,0,0 .

When you run the test, you will now be able to test the border color of the edge of TextField which is actually buried down in the hierarchy of the TextField1.

Index

A

- access 21
- Acrobat and Adobe Reader testing 16
- AcroField 14
- Acroform field properties 20
- afpld 20
- afpModel 20
- afpType 20

B

- borderColor 21
- borderWidth 21
- Browser 5
- Browser viewer testing 17
- Button 14

C

- CheckBox 14
- Checkpoint properties 20
- ClearEvents 17
- Click 17
- Client software 4
- ComboBox 15
- Creating a new test script 15

E

- EnableEvents 17
- Event 18
- event 20, 21
- Event testing
 - Selecting the event 19
- Events provided and supported by the PDF Test Toolkit 17
- eventsEnabled 20, 21
- Example QTP test
 - XML form in browser 25
- Example QTP Tests 24

F

- Flex Builder 3.0 6
- Flex Builder 3.0.2 update 6
- Flex plugin directory 7
- Flex QTP plugin installer 7

G

- Generic properties not subject to user customization 20
- GetEvent 18
- Guidelines for Creating Tests 14

H

- h 21

- Hardware 5
- highlight 21

I

- Information only
 - Registry changes 11
- Install components to support Flex testing 6
- Install Flex Builder 6
- Install Quick Test Professional 8
- Install the Adobe PDF Test Toolkit 5
- Install the Flex plug-in for Quick Test Pro 6
- Install the PDF viewing client 5
- Installation 5
- Installation and Configuration 4

L

- Learning objects in the object repository 21
- ListBox 15
- LiveCycle testing 15

M

- Miscellaneous FAQs and operational details 22

N

- numEvents 21

O

- Operating system 5

P

- PDF Document 14
- presence 21

Q

- QTP
 - Add-In Manager 10
 - Additional installation components 9
 - Custom installation 9
 - Missing Flex components warning 7
- QTP configuration
 - Invoking the standalone PDF viewer 16, 17
 - Pointing to the LiveCycle automation URL 16

R

- RadioButton 15
- RadioButtonList 15
- rawValue 21
- Registry entries 11
- rotate 21

S

- Select 17
- Server software 4
- SetTimeout 18
- SetValue 17
- Signature 15
- size 21
- somExpression 21
- Supported form elements 14
- System requirements 4

T

- Table 15
- TableHeader 15
- TableRow 15
- Test URLs 8
- TestEvents 18
- TestEventsPassed 18
- testEventsPassed 21
- Testing events 17
- Text 15
- timeout 20, 21
- Toggling between forms 22

- Typetext 17

U

- Uninstalling the Adobe PDF Test Toolkit 11

V

- Value 21
- Verify the server (LiveCycle ES) software 8

W

- w 21
- What is the Adobe PDF Test Toolkit? 4

X

- x 21
- XFAForm 15
- XML field properties 20, 21
- XML form in a browser 24

Y

- y 21