



AUTOMATED UI TESTER FOR ADOBE[®] ACTIONSCRIPT[®]

FREQUENTLY ASKED QUESTIONS

Last Updated
30 May 2012

Adobe Systems Inc.

Contents

Contents	2
What is Automated UI Tester for Adobe® ActionScript®?	3
What are the major features of Automated UI Tester for ActionScript®?	3
What is Genie and how it is related to Automated UI Tester for Adobe® ActionScript®?	3
Why is Automated UI Tester for ActionScript better than other Flash automation tools?	3
Does Automated UI Tester for ActionScript work on image comparison or object recognition?	3
Will it be able to identify all the objects uniquely?	3
Is embedding a DLL/SWC, or other library required in an application undergoing testing?	3
Can I use this tool on an IOS or Android device for testing Adobe AIR applications?	3
What platforms and operating systems are supported?	4
What browsers are supported?	4
What kinds of SWF files are supported and what SWF files are not supported?	4
Does it support pure Flash/ActionScript apps?	4
How do I write my script?	4
Will a script created on Windows work on Mac?	4
Can I execute the same script in different languages on a multilingual SWF application?	4
Does Automated UI Tester for Adobe® ActionScript® provide multi-application workflow?	4
How do I get my script execution results?	4
Can I validate my steps in a recorded script?	4

What is Automated UI Tester for Adobe® ActionScript®?

Automated UI Tester for Adobe® ActionScript® is a pure ActionScript 3.0 SWF application automation testing tool with the capability of recording user actions on a SWF file and playing them back with high fidelity. The API for this tool is provided with the name "Genie" (Adobe internal code name for the tool).

What are the major features of Automated UI Tester for ActionScript®?

- No instrumentation or application manipulation is required.
- Automate pure ActionScript 3.0 SWF files.
- No source code access required.
- No need to run inside any wrapper.
- Works on web-based apps and standalone SWF files.
- Performs native automation (no image dependency), thus making scripts reliable and lowering maintenance costs.
- The application can even run in background.

What is Genie and how it is related to Automated UI Tester for Adobe® ActionScript®?

"Genie" is the internal code name for Automated UI Tester for Adobe® ActionScript®. They are referring to the same tool. The API exposed by this tool is called "Genie".

Why is Automated UI Tester for ActionScript better than other Flash automation tools?

With other automation tools, automating applications requires instrumentation of an application or requires running inside a wrapper application. Also, other tools cannot automate pure Flash or ActionScript applications (for example, Flash games), but Automated UI Tester for Adobe® ActionScript® solves all these problems.

Does Automated UI Tester for ActionScript work on image comparison or object recognition?

It works on object recognition. The tool gets the complete object structure of the application under test and provides all objects with unique identifiers in a hierarchical manner to recognize each component individually.

Will it be able to identify all the objects uniquely?

Yes, the tool will recognize the objects uniquely and provide unique identifiers to every object.

Is embedding a DLL/SWC, or other library required in an application undergoing testing?

No. This allows the automation of the current released builds without any recompilation.

Can I use this tool on an IOS or Android device for testing Adobe AIR applications?

No, Automated UI Tester for Adobe Actionscript does not support AIR application on desktop or on any kind of devices.

What platforms and operating systems are supported?

The tool's scripting language is Java; therefore, it supports most platforms: Windows, Mac, Linux, and so on. However, the APIs are best optimized for Windows and Mac.

What browsers are supported?

All browsers that support Flash Player (IE, Firefox, Chrome, Safari, and so on) are supported by this tool.

What kinds of SWF files are supported and what SWF files are not supported?

The tool supports testing on SWF files created in ActionScript 3 that are pure AS applications containing objects like Sprites, movieclips, and such. Not all Flex (MX or Spark) components (Flex apps) are supported; however, Automated UI Tester for Adobe® ActionScript® can be extended to support these. AIR applications, AS1- and AS2-based SWF apps are also not supported.

Does it support pure Flash/ActionScript apps?

Yes, pure Flash/ActionScript apps are supported.

How do I write my script?

The user writes scripts in Java. Automated UI Tester for Adobe® ActionScript® also supports record and playback. Detailed documentation is available at [UserGuide.pdf](#) in the "Script recording and playback" section.

Will a script created on Windows work on Mac?

Script recordings run successfully in multiple platform environments. However script recordings should be played back on the same zoom level.

Can I execute the same script in different languages on a multilingual SWF application?

Yes. The same script will run on different locales of an application, since language-specific information is not used for performing automation.

Does Automated UI Tester for Adobe® ActionScript® provide multi-application workflow?

Yes, multi-SWF app workflows are supported. You can automate a workflow requiring multiple applications in a single script.

How do I get my script execution results?

Automated UI Tester for Adobe® ActionScript® creates a log file in XML format as well as HTML format in a folder with the current date and time. The folder name is labeled with the corresponding date and time from the log file. This is done for every script execution. The log contains the result of each step along with consolidated results. Generally, logs are created in the directory where the Executor.jar file is present (<GeniePath>\GenieScript\GenieLogs folder).

Can I validate my steps in a recorded script?

Automated UI Tester for Adobe® ActionScript® provides a comprehensive set of APIs to validate the steps. Assertion methods for validation are also provided. Detailed documentation is available in

UserGuide.pdf in the “User-exposed methods for validation and generic use” section. Also refer to the JavaDocs of the tool for a full list of exposed APIs.