

Quick Start Guide

for Windows

Examples with Visual Studio & NUnit

Some Useful Definitions

- **AUT**: Application Under Test
- **AUT Launch File**: the configuration file used by the Maveryx Test Automation Framework to run the related AUT
- **Keyword-driven testing**: a codeless approach to write test cases for non-programmers
- **Data-driven testing**: a methodological approach to separate test cases from test data



Requirements

To work with Maveryx, your system shall meet the following minimum requirements:

- Windows 7 or later
- Java Runtime Environment ver. 1.8.0_211 (https://www.java.com/en/download/)
- .NET Runtime Environment ver. 4.6 or later (https://www.microsoft.com/enus/download/details.aspx?id=48130)



Summary

- Install and configure Maveryx
- Get license key
- Run the Demo Solution
- Create and run your first test







- Install and configure Maveryx
- Get license key
- Run the Demo Solution
- Create and run your first test





Installation (1)

To install the Maveryx Test Automation Framework on your system, run *Maveryx_Win_2.X.y_Trial.exe* and follow the steps of the setup wizard.

Choose the directory into which you want to install the Maveryx software. <u>You must</u> <u>have write permissions to this directory</u>.





Installation (2)

You can skip this step but, in case, you can also **Install the Eclipse plug-in** by clicking on the checkbox and selection the Eclipse path.



If you want to install the Maveryx Eclipse Plugin later, copy the files in **MAVERYX_HOME\tools\EclipsePlugin** folder into the **/dropins** directory of your Eclipse installation.



Setting up Visual Studio (1)





Setting up Visual Studio (2)

Language packs

Installation locations

Vieual Studio Installor

Installing — Visual Studio Community 2017 — 15.9.7

Workloads Individual components Visual C++ runtime for UWP

- Visual C++ tools for CMake
- Windows Universal CRT SDK
- Windows XP support for C++

Debugging and testing



- JavaScript diagnostics
- Just-In-Time debugger
- Test Adapter for Boost.Test
- Test Adapter for Google Test
- Testing tools core features

Development activities

	ASP.NET and web development tools
~	C# and Visual Basic
	C++ Android development tools
	C++ iOS development tools
	Cookiecutter template support
1	Fache dated and lat Development

Location

C:\Program Files (x86)\Microsoft Visual Studio\2017\Community Change...

By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software w is licensed separately, as set out in the <u>3rd Party Notices</u> or in its accompanying license. By continuing, you also agree to those licenses.



Select the Visual Studio individual components as in the example



- Install and configure Maveryx
- Get license key
- Run the Demo Solution
- Create and run your first test





Maveryx License

- Maveryx uses a **Node-Locked** license type. A node-locked license for Maveryx lets you run the application on a specific machine or workstation. This license type is considered a single-user license, although it's bound to the machine, not the user.
- Trial versions of Maveryx (which is always licensed as Node-Locked) have a time-limited license. After it expires, you can no longer use the product.
- After a commercial license for Maveryx expires, you can continue using the product. However, you will not be able to get updates for the product and technical assistance from the Maveryx Support team.



Collecting Node Data

To generate a license open MAVERYX_HOME/bin/ folder and run the LicenseGen.bat file.

This utility will automatically collect all hardware and software information needed to generate a valid Maveryx license by saving them into the **config.properties** file in **MAVERYX_HOME/bin/** folder.

📕 🚽 📃 🖛 🛛 bin	Strumenti applicazioni	- 🗆 ×	📙 📝 📙 🖛 bin		- C	X	
File Home Condi	ividi Visualizza Gestisci	~ (File Home Condi	ividi Visualizza		~ 🤇	
Aggiungi ad Accesso rapido	J K Sposta in v K Elimina v J S Sposta in v Rinomina Nu Copia in v Rinomina	tella	Aggiungi ad Accesso rapido	Ia Ia Ia Ia Ia Ia Ia Copia in ▼ X Elimina ▼ Ia Rinomina	Nuova cartella	utto a tutto zione	
Appunti	Organizza	Nuovo Apri Seleziona	Appunti	Organizza	Nuovo Apri Seleziona	a	
← → × ↑ 📙 « Die	scolocale (C:) → Maveryx → bin	✓ ♂ Cerca in bin	\leftarrow \rightarrow \checkmark \uparrow \square \ll Dis	sco locale (C:) \rightarrow Maveryx \rightarrow bin \rightarrow	マ ひ Cerca in bin	Q	
Accesso rapido	log plugin	Remote_Server_START.bat	Accesso rapido	log plugin	proxyAdvanced.jar Remote_Server_START.bat		
ConeDrive	remote report	i remote>erver.jar i tEnv.jar i SearchingStrategyPlugInSettings.xml	Cuesto PC	remote report	its remoteServer.bat i remoteServer.jar i i tEnv.jar		
Documenti	winViewerEnv	testReportTemplate.xml	Documenti	winViewerEnv	TemplateMatching.properties		
🖊 Download 📰 Immagini	💿 AUTConfiguratorUI.bat 🍙 autConfiguratorUI.jar	ver.bxt	🔶 Download 📰 Immagini	💿 AUTConfiguratorUI.bat 🍙 autConfiguratorUI.jar	i testReportTemplate.xml i ver.bt		
🍌 Musica 🧊 Oggetti 3D	الله keydrivenTesting.jar الله KeywordDrivenTestingCLI.bat	_	Musica Oggetti 3D	config.properties			
Video	LicenseGen.bat maveryx.exe		Video	KeywordDrivenTestingCLI.bat LicenseGen.bat			
Disco locale (C:)	📓 MaveryxSetup.xml 遙 proxyAdvanced.jar		Disco locale (C:)	📧 maveryx.exe 📝 MaveryxSetup.xml			
23 elementi 1 elemento	selezionato 32 byte		24 elementi 1 elemento	selezionato 4,16 KB			



License Key

When you will have the **config.properties** file, send it by email to **license.manager@maveryx.com** with the subject "License Key Request".

In reply to your email, you will receive your license key file (license.dat) as attachment.

Save the license.dat file into the MAVERYX_HOME/bin/ folder.







- Install and configure Maveryx
- Get license key
- Run the Demo Solution
- Create and run your first test





Open the Demo Solution (1)

Microsoft Visual Studio

File	Edit View Project Debu	ig Team	Tool	s -	Test Analy	ze	Wind	low	Help	
	New		٠				- >	Atta	ach	1 🔎 🚽
	Open			雷	Project/Sol	ution			c	trl+Shift+O
œ	Start Page			2	Folder			C	Ctrl+Shift+Alt+O	
	Close				Open from	Sour	ce Cor	ntrol		
	Close Solution			2	File				c	trl+0
ыi	Save Selected Items	Ctrl+S			Convert					
	Save Selected Items As									
2	Save All	Ctrl+Shift+	S							
	Source Control		•							
	Page Setup				امک	<u>م</u> م	+ 6	ilo		
.	Print	Ctrl+P				こし	ι.		-	
	Account Settings				Pro	je	ct/	Sc	olut	ion
	Recent Projects and Solutions					_				
	Exit	Alt+F4								



Open the Demo Solution (2)

The solution file NUnitTest.sln is located into the **MAVERYX_HOME\demo\.NetDemo\NunitTest** folder.

ganiz	e 💌 New folder					8== 🖕	
ganizi		~	Discourse Differentia	T		S *	
	Name		Date modified	lype		Size	
2	.NET		2/12/2019 5:57 PM	File fo	lder		
	.vs		2/12/2019 5:57 PM	File fo	lder		
12	📙 Java		2/12/2019 5:57 PM	File fo	lder		
	packages		2/12/2019 5:57 PM	File fo	lder		
4	Properties		2/12/2019 5:57 PM	File fo	lder		
•	Web		2/12/2019 5:57 PM	File fo	lder		
	C ■ NUnitTest.csproj		2/12/2019 5:57 PM	Visual	C# Project f	7	KB
E	State NUnitTest.sln		2/12/2019 5:57 PM	Visual	Studio Solu	2	KB
•							
	File <u>n</u> ame: [NUnitTest. <mark>s</mark> In		~	All Project Fi	les (*.sln;*.	dsw;*.vc
					Open		Cancel



The Demo Solution (1)



.NET

The built-in Demo project has many "ready to be executed" examples for .NET Desktop Applications.



The Demo Solution (2)



Java

The built-in Demo Solution has many "ready to be executed" examples for Java Desktop Applications.



The Demo Solution (3)



Test it simple!

Web

The built-in Demo project has many "ready to be executed" examples for Web Applications.



Run a Test Script



Test it simple!

In the Test Explorer window, right click on the test class or test case you want to run



- Install and configure Maveryx
- Get license key
- Run the Demo Solution
- Create and run your first test





Create and run your first test

1. Create a new Maveryx Test Project

- 2. Create a new Maveryx Test Class
- 3. Create the AUT Launch file
- 4. Write the test case
- 5. Run the test



Create New Test Project (1)

New Project					? ×
♦ Recent		Sort by:	Default 🔹 🏭	Ē	Search (Ctrl+E)
 Recent Installed Visual C# Get Started Windows Desktop .NET Standard Test Visual Basic Other Project Types Online 			WPF App (.NET Framework) Windows Forms App (.NET Framework) Console App (.NET Framework) Class Library (.NET Standard) Class Library (.NET Framework) Shared Project Class Library (Legacy Portable)	Visual C# vrk) Visual C# Visual C# Visual C# Visual C# Visual C# Visual C# Visual C#	Type: Visual C# A project for creating a C# class library (.dll)
Not finding what y Open Visual S	ou are looking for? tudio Installer				
<u>N</u> ame:	PasswordDemoTest				
Location:	C:\Users\Mauro\sou	rce\repos	έ.		Browse
Solution:	Create new solution			-	
Solution name:	PasswordDemoTest				Create directory for solution
<u>F</u> ramework:	.NET Framework 4.6.	2 -]		Add to So <u>u</u> rce Control OK Cancel

- 1. Select File \rightarrow New \rightarrow Project...
- In the **New Project** window
 - 1. Select Class Library(.NET Framework) Visual C#
 - 2. Enter the Name and Solution name (e.g. "PasswordDemoTest")
 - 3. In the **framework** section make sure that .NET Framework 4.6.2 is selected
- 2. Click OK



Create New Test Project (2)





Create New Test Project (3)





Create and run your first test

1. Create a new Maveryx Test Project

2. Create a new Maveryx Test Class

- 3. Create the AUT Launch file
- 4. Write the test case
- 5. Run the test



Create New Test Class (1)

Add New Item - PasswordDemoTest

 Installed 		Sort by:	Default	· # E		Sei	arch (Ctr	I+E)	- م	
 Visual C# Items WPF 		[Class		Visual C# Items		ype: Vis	sual C# Items / class definition		
Code Data		••	Interface		Visual C# Items					
General Windows Forn	ns	==	Windows Form		Visual C# Items	L	Sel	ect Projec t	$t \rightarrow Ade$	d
▶ Online		•	User Control		Visual C# Items		Cla	ISS		
		£	Component Class		Visual C# Items		In 🖌	Add New It	em win	dow
			User Control (WPF)		Visual C# Items		1.	Select Vis	ual C#	Item
		v.1.0	About Box		Visual C# Items		2.	Select Cla Enter a Na	i ss ame for	. the
		Ŷ	Application Configuration	File	Visual C# Items		0.	test class	(exam	ple
			Application Manifest File		Visual C# Items			MyTest.cs	3)	
		Ĩ	Assembly Information File		Visual C# Items		4.	Click Add		
		\approx	Bitmap File		Visual C# Items					
		Ŷ	Code Analysis Rule Set		Visual C# Items	-				
<u>N</u> ame:	MyTest.cs									
								Add	Cancel	



© 2021 Maveryx srl. All rights reserved.

?

× ۵.

C# Items

Create New Test Class (2)

using com.maveryx.bootstrap; using NUnit.Framework;

⊡namespace NUnitTest.Java

[TestFixture] public class TestClass

string _autPath= "C:/Maveryx/demo/AUT/PasswordDemo.xml";

[SetUp] public void SetUp()

Bootstrap4Net.Start();

Bootstrap4Net.StartApplication(_autPath);

[TearDown] public void TearDown()

Bootstrap4Net.Stop();

[Test] public void Test001()

//write here yout test script

Path to the AUT xml file

Start the Maveryx

Start the AUT

Stop the Maveryx Framework



Create and run your first test

- 1. Create a new Maveryx Test Project
- 2. Create a new Maveryx Test Class

© 2021 Maveryx srl.

All rights reserved.

3. Create the AUT Launch file

- 4. Write the test case
- 5. Run the test



Java AUT Launch File

To execute a Java Application-Under-Test it is necessary to create the related AUT launch file.

<?xml version="1.0" encoding="UTF-8"?>

```
<AUT_DATA>
```

<SERVER_URL></SERVER_URL>

<WORKING_DIR>./src/resources/AUT/java</WORKING_DIR> <!-- change this path to your working directory -->

<APPLICATION_NAME>ButtonDemo</APPLICATION_NAME>

<AUT_ARGUMENTS></AUT_ARGUMENTS>

<VM_ARGUMENTS></VM_ARGUMENTS>

<DESCRIPTION>

Push-Button testing </DESCRIPTION>

<JRE_PATH>\${java.home}</JRE_PATH> <!-- change this path to your JRE home -->

<MAIN_CLASS>com.sun.demo.ButtonDemo</MAIN CLASS>

<!-- on UNIX-like and MAC OS X systems change the path separator ';' to ':' --> <CLASSPATH>

<LIB>

<PATH>examples.jar</PATH> <!-- change this path to your Maveryx installation directory /demo --> </LIB>

<!-- do not change the data below! (except for path separator on UNIX-like and MAC OS X systems) -->
</CLASSPATH>

</AUT_DATA>



MFC & .Net AUT Launch File

To execute a MFC or .NET Application-Under-Test it is necessary to create the related **AUT launch file**.

Set the absolute or relative path to your AUT executable file



Web AUT Launch File

To execute a Web Application-Under-Test it is necessary to create the related **AUT launch file**.



Create and run your first test

- 1. Create a new Maveryx Test Project
- 2. Create a new Maveryx Test Class
- 3. Create the AUT Launch file
- 4. Write the test case
- 5. Run the test



Example

[Test] public void Test001()

```
var t = new GuiPasswordText("Enter the password");
//check wheter the text field is editable
Assert.True(t.IsEditable());
//enter the password
t.SetText("bugaboo");
```

```
var ok = new GuiButton("OK");
//check wheter the push button is enabled
Assert.True(ok.IsEnabled());
//click the OK button in the main frame
ok.Click();
```

```
var dialog = new GuiDialog("Message");
var message = new GuiLabel("Success!", dialog);
```

```
var expectedMessage = "Success! You typed the right password.";
//check wheter the message dialog contains the excepted message
Assert.True(message.GetActualId().Equals(expectedMessage));
```

//close the dialog
dialog.Close();

MANERY

Test it simple!

PasswordDemo
Enter the password: OK Help
PasswordDemo
Enter the password: ••••••• OK Help
Message 🛛
i Success! You typed the right password.
© 2021 Maveryx srl All rights reserved

Create and run your first test

- 1. Create a new Maveryx Test Project
- 2. Create a new Maveryx Test Class
- 3. Create the AUT Launch file
- 4. Write the test case
- 5. Run the test



Run the Test Script





In Test Explorer window, right click on the test class or test case you want to run

© 2021 Maveryx srl.

All rights reserved.

sales@maveryx.com info@maveryx.com +39 333 30 72 597 +39 351 87 85 706

 \bigcirc

