

Hands-On Lab

Setup Microsoft Test Manager 2012 and froglogic Squish for Automated GUI Testing

Lab version: 1.0.2

Last updated: 27/03/2013



Overview

This hands-on lab is part one out of a series of two labs showing how to use Microsoft Visual Studio 2012 and Microsoft Test Manager 2012 to run an automated Squish GUI tests.

Squish is a cross-platform, cross-device GUI test automation tool allowing to automate GUIs on a variety of platforms based on several different GUI technologies. The Squish for MS ALM plugin allows to centrally manage Squish GUI tests in TFS and to run them from Visual Studio, via TFS scheduled builds and continuous integration and via Microsoft Test Manager.

In this first lab, you will learn how to install Squish GUI Tester and how to create a Microsoft Test Manager 2012 test environment to integrate Squish. To learn how to finally run the test from Microsoft Visual Studio 2012 and as part of a Microsoft Test Manager 2012 test plan, please refer to the second lab of this series: [Authoring and Running Automated GUI Tests using Microsoft Test Manager 2012 and froglogic Squish](#).

This hands-on lab is one out of a number of labs that deal with Microsoft Test Manager 2012. It focuses on running automated tests with Squish GUI Tester. To learn more about Microsoft Test Manager 2012 features that are unrelated to Squish GUI Tester, please refer to this [this blog post](#).

Prerequisites

In order to complete this lab you will need the Visual Studio 2012 virtual machine provided by Microsoft. For more information on acquiring and using this virtual machine, please see [this blog post](#).

Exercises

This Hands-On Lab comprises the following exercises:

1. Installing froglogic Squish GUI Tester
 2. Create a Microsoft Test Manager 2012 test environment
-

Estimated time to complete this lab: **30 minutes**.

Exercise 1: Installing froglogic Squish GUI Tester

In this exercise, you will learn how to install Squish GUI Tester to be able to author automated GUI tests.

1. Download the Squish ISO image from <http://download.froglogic.com/resources/squish-vs2012-hyperv-vm-latest.iso>.
2. Contact sales@froglogic.com to let them provide you with an evaluation key. This Lab uses Squish for Java, which is capable of testing Java applications. The Squish installer contains other editions as well, for example, to test native Windows applications, iOS applications or application based on various other technologies. If you are interested in trying out those editions together with Microsoft Visual Studio 2012 or Microsoft Test Manager 2012, feel free to ask for further evaluation licenses.
3. Start the virtual machine. Open **Media | DVD Drive | Insert Disk...** Browse for the downloaded ISO image and select it.
4. Microsoft Windows will display an **AutoPlay** dialog. Select **Run Setup.exe**.

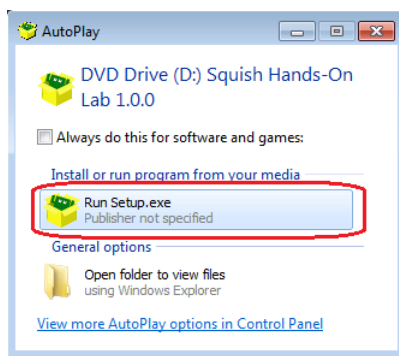


Figure 1
Start the Squish setup via AutoPlay

5. Click on **Next** to start the installation procedure and accept the license.

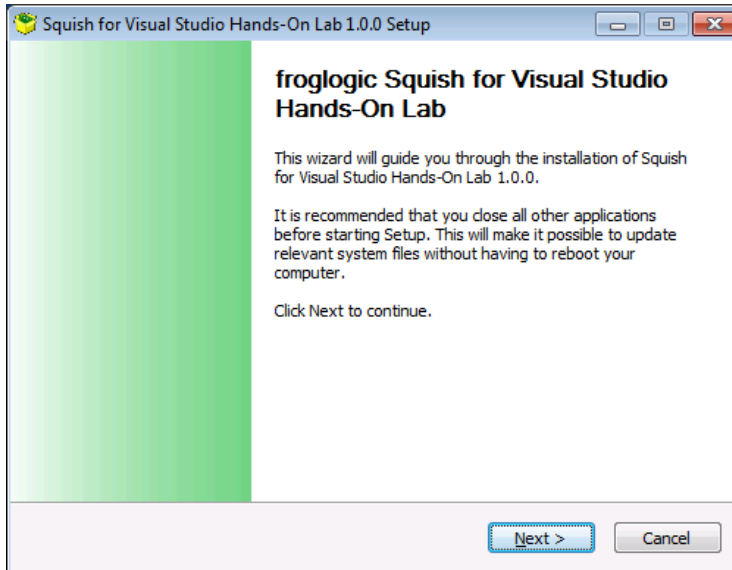


Figure 2
Squish Installer

6. In the next step, select the **Squish for Java** as the Squish edition to install and click on **Next**. You can also select different editions but this Lab explains how to use the one for Java. Note that you can re-run the Squish installer at any time to install further editions.

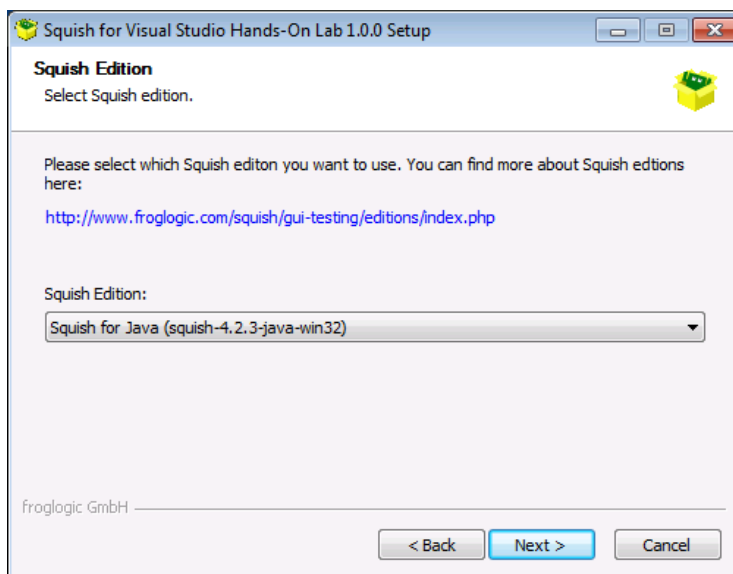


Figure 3
Selection of the Squish edition

7. Enter the **Squish License Key** that has been provided by sales@froglogic.com and click on **Install**. The installation process can take some minutes. Internet access is not required for this step.

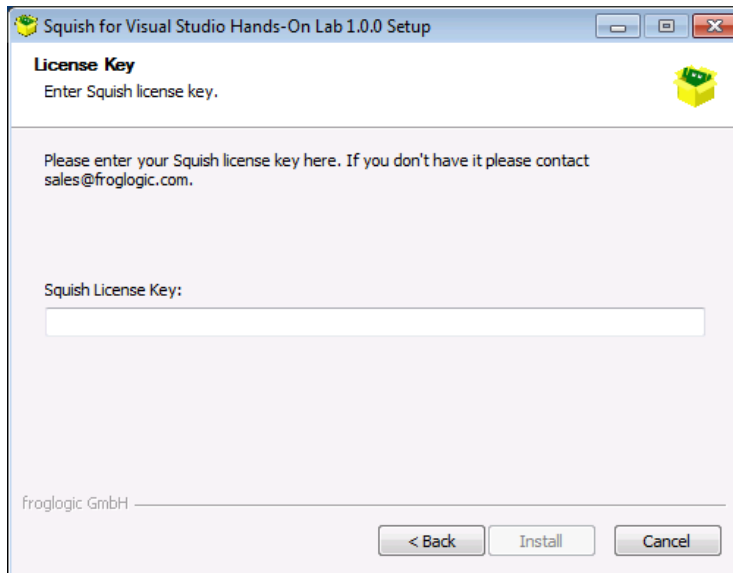


Figure 4
Entering the Squish license key

8. Finally, click on **Finish**. You will find Squish shortcuts on the desktop. The shortcut **Squish for Java** starts the Squish IDE for test authoring. The shortcut **Eclipse with Squish** opens the Eclipse IDE with Squish being integrated. The shortcut **Squish VS Example** opens an example project in Visual Studio. For this and the following lab, only **Squish for Java** is required.

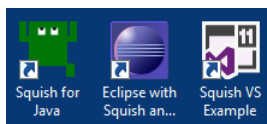


Figure 5
Squish desktop icons after installation

In the next exercise, you are going to setup Microsoft Test Manager 2012 to be able to run Squish GUI tests from it.

Test authoring with Squish and test execution with Microsoft Test Manager 2012 is finally covered in the second hands-on lab of this series: Authoring and Running Automated GUI Tests using Microsoft Test Manager 2012 and froglogic Squish.

Exercise 2: Create a Microsoft Test Manager 2012 test environment

In this exercise, you will learn how to prepare Microsoft Test Manager 2012 to run automated GUI tests that have been created with Squish. You are going to create a new test environment in Microsoft Test Manager 2012 and install and configure a Visual Studio Test Agent.

1. For an automated test run, we need to specify the machine on which the test should be run on. This is done by configuring a test environment. To start with this, open **Start | All Programs | Microsoft Visual Studio 2012 | Microsoft Test Manager**.
2. Microsoft Test Manager will connect to the most recently loaded team project, which in the case of this virtual machine is the project named **Tailspin Toys**. As you have to open a different team project for this lab, click on the Tailspin Toys link in the upper-right corner.

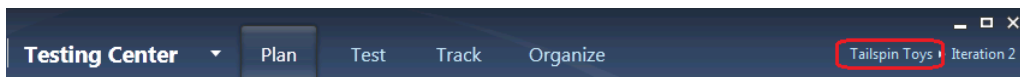


Figure 6

Currently opened team project

3. Select **vsalm | FabrikamFiberCollection | FabrikamFiber** and click on **Connect now**. In the dialog that pops up, confirm to connect to the team project.

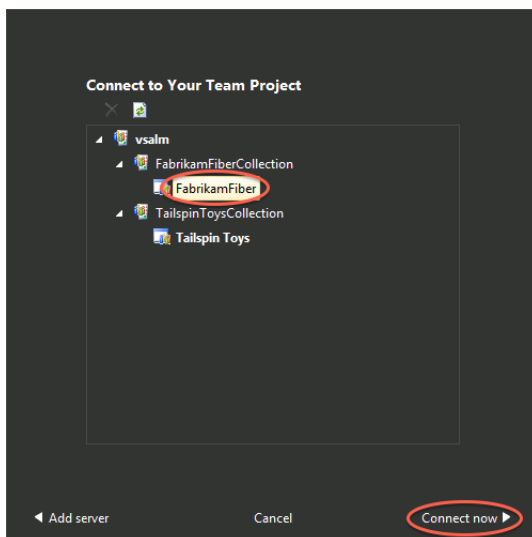


Figure 7

Location of the Team Project and Connect now button

4. Select the **Default** test plan and click on **Select plan**.

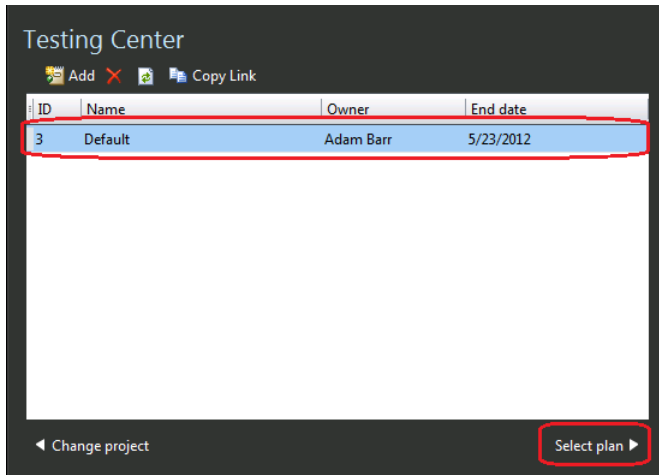


Figure 8

Selecting the test plan

5. Click on **Testing Center** and select **Lab Center**. Click on the **New** button to create a new environment.

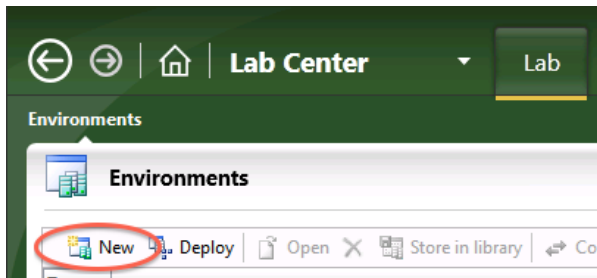


Figure 9

Creating a new environment

6. This will open the configuration view for the environment. In the **Type and name** section, enter **SquishEnvironment** into the field **Name** and click on **Next**.

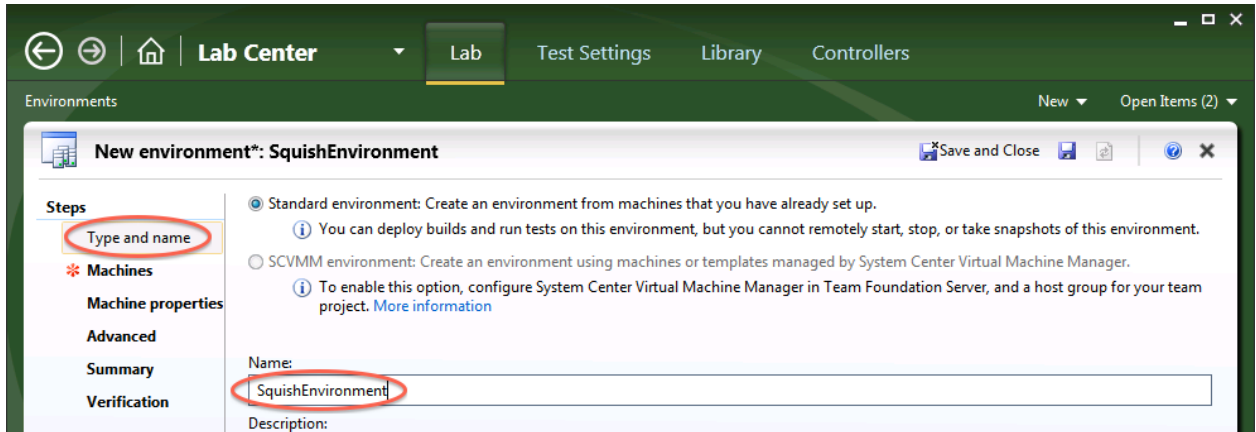


Figure 10
Setting a name for the environment

- In the **Machines** section, click on the **Add machine** button. Enter **VSALM** for the computer name. This is the name of the virtual machine you are using. Open the **Select/Type** role combo box and select **Desktop Client**. In the field **User name**, enter **.\Julia**. The **password** should be **P2ssw0rd** again.

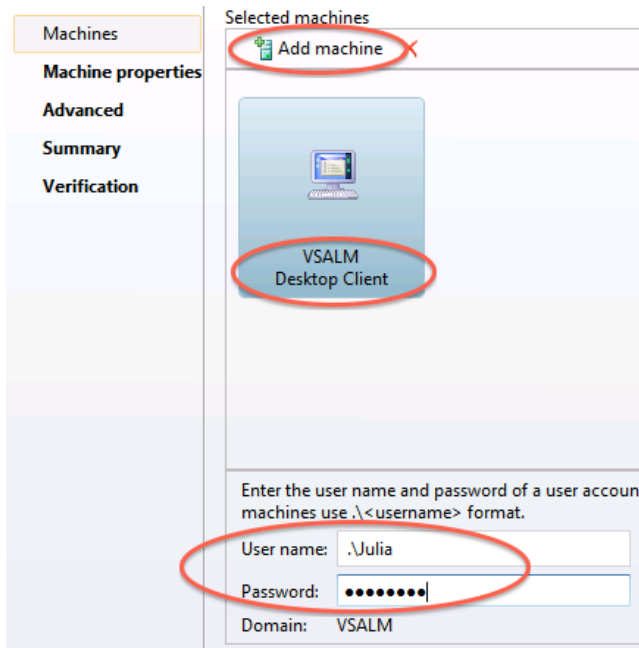


Figure 11
Configuration of the machine for the automated test run

Note: Do not configure anything else than the Machines section.

- Click **Verify** and wait for the verifications to be finished.

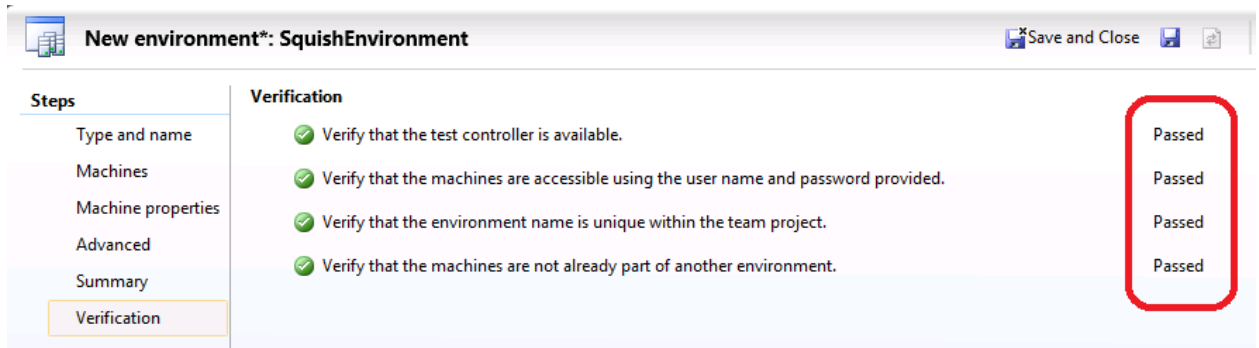


Figure 12
Result of the environment settings' verification

- Finally, click on **Finish**. This will start the installation process for Visual Studio Test Controller and Agent. This may take a couple of minutes.

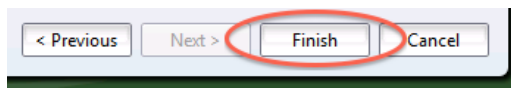


Figure 13
Location of the Finish button to start Controller and Agent installation

- Wait until the environment is listed as **Ready**.

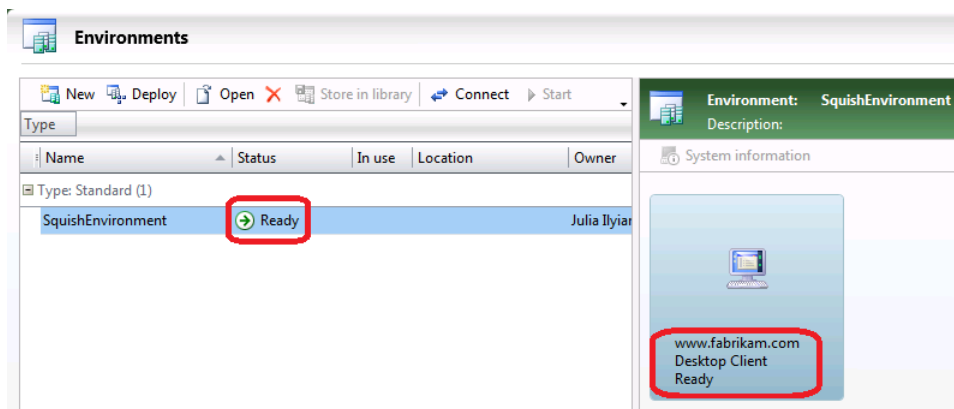


Figure 14
Test environment status

- As part of the test environment, Microsoft Test Manager installed a Visual Studio Test Agent. This will be used for the automated tests runs later on. So we need to do an additional configuration step: the Test Agent needs to be able to run applications with a GUI. To configure this, open **Start | All Programs | Microsoft Visual Studio 2012 | Test Agent Configuration Tool**.

- Click on the **Run Options** button and select **Interactive process**. Click **Next** to go back to the main screen.

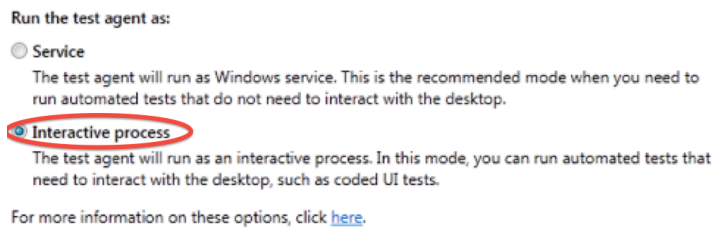


Figure 15

Settings to start the Visual Studio Test Agent as an interactive process

- Finally, in the **Run agent as an interactive process** section, enter **.\Julia** as the **User name** and **P2ssw0rd** into the **Password** field.

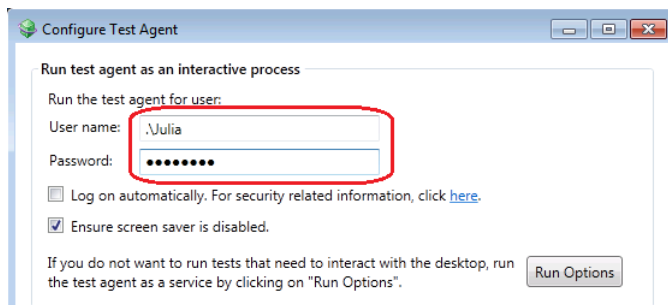


Figure 16

Entering user name and password for the Test Agent

- Click on the **Apply Settings** button and wait for the configuration to be finished.

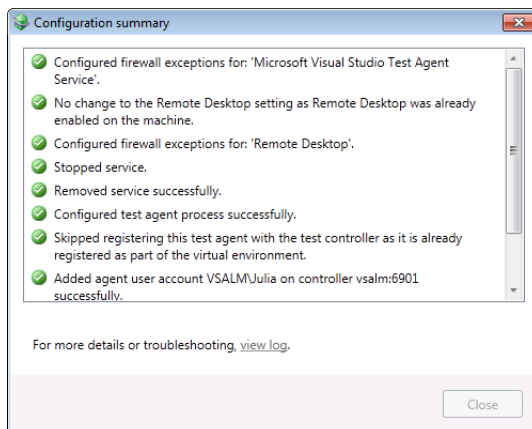


Figure 17

Configuration progress dialog

15. Once the configuration of the Test Agent is finished, the Test Agent Status window should display the status **Online**.

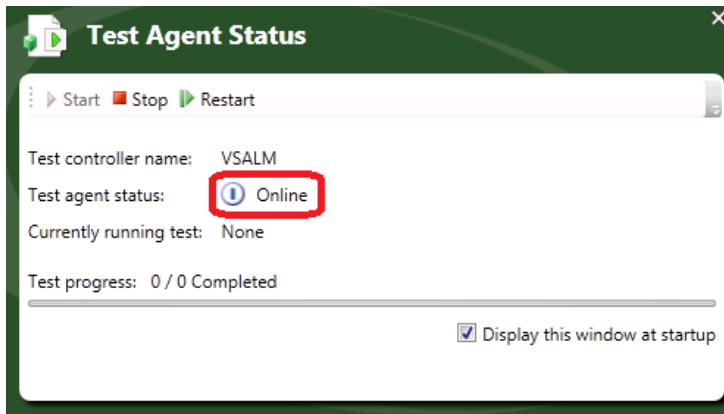


Figure 18
Visual Studio Test Agent Status window

16. After creating the test environment, Microsoft Test Manager will be able to execute automated Squish GUI tests. To learn how to do this, please refer to the next hands-on lab of this series: Authoring and Running Automated GUI Tests using Microsoft Test Manager 2012 and froglogic Squish.

To give feedback please write to sales@froglogic.com

Copyright © 2013 by froglogic GmbH. All rights reserved.