

TARGETPROCESS SELENIUM AND NUNIT INTEGRATION GUIDE



v.2.13

Selenium and NUnit Integration Guide

This document describes Selenium and NUnit plugin configuration.

SELENIUM/NUNIT INTEGRATION OVERVIEW	2
CONFIGURE TARGETPROCESS NUNIT INTEGRATION PLUGIN.....	3
STEP 1. TEST CASES TO TESTS MAPPING.....	3
STEP 2. TEST PLAN CREATION	4
STEP 3. PROFILE CONFIGURATION	4
CONFIGURE TARGETPROCESS SELENIUM INTEGRATION PLUGIN.....	6
STEPS 1-2.	6
STEP 3. CONFIGURE PROFILE.....	6
STEP 4. CONFIGURE SELENIUM	7
OTHER.....	8

Selenium/NUnit Integration Overview

Selenium/NUnit integration imports test results into TargetProcess, associates test results with existing test plan and creates new test plan run. It allows you to have daily builds results in TargetProcess, providing real time information about product quality in a single place.

The process consists of several steps:

1. Map tests in NUnit or Selenium with Test Cases in TargetProcess. You may use name conventions and regular expressions to build the mapping. As a result you should have set of test cases in TargetProcess and set of related tests in NUnit or Selenium.
2. Create Test Plan and assign required test cases to this test plan.
3. Specify NUnit (or Selenium) test results XML file location.
4. Execute tests in NUnit
5. Plugin will get the test results, parse them, create new test run and add test run results into the system.

Also you may post test results into special URL in TargetProcess from Selenium.

Configure TargetProcess NUnit Integration Plugin

Step 1. Test Cases to Tests Mapping

You have a set of tests in NUnit. To have test results in TargetProcess, it is required to have a set of related test cases. For example, you have 3 tests in NUnit:

```
AddSimpleUserTest
AddAdminTest
EditAdminTest
```

You want to run them and have results available in TargetProcess.

First it is required to map these tests to TargetProcess test cases. There are several options. The simplest option is just have 3 test cases in TargetProcess with the same names or with spaces, like this:

NUnit Test	TargetProcess Test Case
AddSimpleUserTest	Add Simple User Test
AddAdminTest	Add Admin Test
EditAdminTest	Edit Admin Test

The other option is to add test case id to each test in NUnit. And use regular expression to extract the id from the test name `_(?<testId>\d+)$`

If you are not familiar with regular expressions, it may be hard to write correct one. Please check some links in the end of the document.

```
[Test]
public void AddSimpleUserTest_90()
{
    Navigation.Login();
    CurrentEntityOperator.Retriever.SetCurrentEntity(Creator.Project());
    ...
}
```

NUnit Test	TargetProcess Test Case
AddSimpleUserTest_90	Add Simple User Test
AddAdminTest_91	Add Admin Test
EditAdminTest_95	Edit Admin Test

Alternatively you may cut part of the name, for example, remove Test word from test names using regular expression `^(?<testName>[^_]+)Test$`

NUnit Test	TargetProcess Test Case
AddSimpleUserTest	Add Simple User
AddAdminTest	Add Admin
EditAdminTest	Edit Admin

If you do not have test cases, you may add them manually or **import from CSV file**.

Step 2. Test Plan Creation

When you have a list of test cases in TargetProcess, it is required to assign them to test plan. Test Plan is just a set of test cases that may be executed and tracked.

Click to Add link and select Test Plan option. Specify test plan name and save it. Then go to Test Plans list, find required test plan, click Actions link and select Assign Test Cases action.

Assign required test cases to test plan.

Step 3. Profile Configuration

Navigate to [Admin](#) -> [Plugins](#) and click *Add Profile* for NUnit Integration plugin. The profile form will appear. You should specify Profile Name. It is recommended to use project name in profile name, since each profile relates to exactly one project.

Automatic NUnit Test Run Import

Imports automatic test run results to TargetProcess from NUnit.

Profile Name

NUnit

Test Result File Path

Path to the file that contains test results. Ex: *C:\nunit\results.xml*,
http://hostname/results.xml,
ftp://login:password@hostname/results.xml.

Integration Interval

Test results will be imported with defined time interval in hours. Ex:
24.

Test Case

Regex

Use regular expression to find test names. Ex: *^(?<testName>[^_]+)_Test\$* will extract test case name from
ComplexFeature_Test_(?<testId>\d+)\$ will extract test case id from
ComplexFeature_1289.

Test Plan

Test Plan ID

Test automation results bind to the test plan.

Profile parameters:

- **Test Results File Path:** NUnit should be configured to put XML test results into location accessible by TargetProcess. It may be local or network folder, HTTP or FTP path. FTP path should contain login and password in the string. For example, *ftp://mylogin:complexpassword@ourserver.com/nunit/results.xml*
- **Integration Interval:** TargetProcess will check results using this setting. For example, you may check results each day and specify 24.
- **Test Plan:** test results will be associated with the test plan.
- **Mapping Template:** regular expression that powers mapping between NUnit tests and TargetProcess test cases.

Click Save button and then enable profile.

Acceptance Testing

Automatic NUnit Test Run Import
Imports automatic test run results to TargetProcess from NUnit. [Add Profile](#)

Smoke Tests

Automatic Selenium Test Run Import
Imports automatic test run results to TargetProcess from Selenium. [Add Profile](#)

Configure TargetProcess Selenium Integration Plugin

Steps 1-2.

These steps are the same as for NUnit integration. Please check them above.

Step 3. Configure Profile

Profile configuration is similar to NUnit integration profile. It is required to specify profile name, test plan and test mapping expression.

Selenium integration has two modes:

1. **Extract result** from file (same way as NUnit integration works)
2. **Post results** to special URL in TargetProcess (**TestSuiteCallback.ashx**)

If you want to extract results from file, configure path to results file same way as for NUnit integration.

If you want to post results to TargetProcess, just leave Path field empty.

Automatic Selenium Test Run Import

Imports automatic test run results to TargetProcess from Selenium.

Profile Name

Result File

Test Result File Path

Path to the file that contains test results. Ex: C:\nunit\results.html, http://hostname/results.html, ftp://login.password@hostname/results.html.

Passive Mode

Use passive mode for transfers over FTP protocol.

 False

Integration Interval

Test results will be imported with defined time interval in hours. Ex: 24.

Test Plan

Test Plan ID

Test automation results bind to the test plan.

Leave empty if you want to post test results from Selenium

Test Case

Mapping Template

Use regular expression to find test names. Ex: `^(?<testName>[^_]+)_Test$` will extract test case name from `ComplexFeature_Test`, `_(?<testId>[d+])$` will extract test case id from `ComplexFeature_1289`.

See next step to configure Selenium.

Step 4. Configure Selenium to Post Results

Selenium does not store test results in a file, but invokes the URL in TargetProcess instead and post the results to this URL.

When you run Selenium via web browser, you should specify **resultUrl** parameter.

NOTE: Make sure that **auto** is set to **true**.

file:///D:/Selenium/core/TestRunner.html?test=...&auto=true&resultsUrl=http%3a%2f%2flocalhost%2fTargetProcess2%2fSelenium%2fTestSuiteCallback.ashx%3fprofile%3DSmokeTests

Note that resultUrl parameter is **encoded**. The test run above will post results to

http://localhost/TargetProcess2/Selenium/TestSuiteCallback.ashx?profile=SmokeTests

You may form the required URL using pattern `http://<path to TargetProcess>/Selenium/TestSuiteCallback.ashx?profile=<plugin profile name>`. And replace all special characters like `/` or spaces with encoded values.

Then you will just run tests and test results will appear in TargetProcess.

The screenshot shows the Selenium TestRunner interface. The main window displays a list of test cases for 'DogfoodTestRefreshFrame'. A red arrow points to the console log at the bottom, which shows the following text:

```

Importing test results...
Done importing test results.
Created test plan run ID: 7041
Imported test cases:
  Test Open
  Test Click
Not imported test cases:
  TestUseXPathLibrary
  Test Browser Version
    
```

The Selenium TestRunner panel on the right shows the following summary:

- Elapsed: 00:49
- Tests: 72 run, 579 passed, 5 failed, 1 incomplete
- Commands: 11 failed

Other

Regular expressions:

<http://www.radsoftware.com.au/articles/regexlearnsyntax.aspx>

<http://msdn.microsoft.com/en-us/library/aa719739.aspx>

Please, email to **support@targetprocess.com** if any problems with JIRA integration.

Visit support forum at <http://support.targetprocess.com> or TargetProcess Help Desk at

<http://helpdesk.targetprocess.com>

If you have any questions about TargetProcess, just contact us:

Email: info@targetprocess.com

Web: <http://www.targetprocess.com>