

## 02-TestCases.xls

### Purpose of the tool

If you want to create a lot of test cases in TestLink easily, the easiest way is to use tables.

TestLink is able to use XML to import test cases. This document explains how to import Test cases into TestLink using XML files. Excel is used as editor to add new test cases.

### Works with

- ✓ TestLink 1.9.3 → TestLink 1.9.17
- ✓ Attachments are handled through XML files since TestLink 1.9.17 – TestLink 1.9.18 preconized



### Summary

Purpose of the tool.....	1
Works with .....	1
Summary .....	1
Test cases declaration using Excel.....	2
XML file generation .....	6
Test cases Import into TestLink.....	8
Import into Excel TestCases Exported from TestLink.....	10
Tips and problems .....	13

## Test cases declaration using Excel

To declare test cases, the excel pattern named “02-TestCases.xls” must be used. This file could be used to update existing tests or to create new tests.

### Excel pattern overview

Test Suite		 Import Test Cases From TestLink	Test Case		Export Test Cases To TestLink 	Attachments	Steps		Coverage
Name	Details	Name	Summary	PreConditions	Number of Attachments	Step	Expected Result	Document ID	
Parent myTestSuite	Details about the parent test suite Details about my TestSuite	A first TestCase	The summary of the first TestCase			The first Step	Expected result for the first step	Req Sandbox-1	
		A second TestCase	The summary of the second TestCase	A precondition is defined		A second Step	And an expected result for the second step	Req Sandbox-3	
		A second TestCase	A second test case for my TestSuite			A second Step	Expected result for the first step And an expected result for the second step		
Parent YourTestSuite	Details about the parent test suite Details about your TestSuite	Your first TestCase	The summary of the first TestCase of your testSuite	Your precondition		The first Step	Expected result for the first step	Req Sandbox-4	
		A second TestCase	The summary of the second TestCase			A second Step	And an expected result for the second step		
		Your second TestCase	A second test case for Your TestSuite			The first Step A second Step A third Step	Expected result for the first step And an expected result for the second step Expected result for the third step	Req Sandbox-4 Req Sandbox-2	

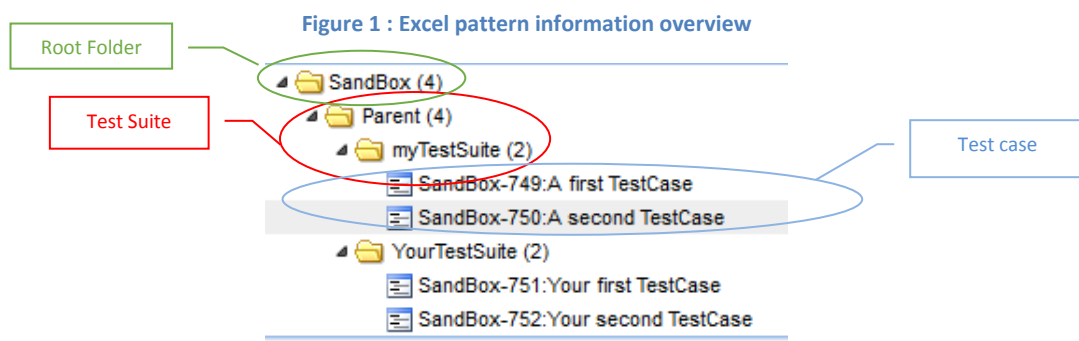


Figure 2 : imported results overview

- The “Test Suite” section (dark blue columns header) can be used to declare folders trees in TestLink (folders are named “Test Suite Operations” in TestLink GUI). The TestCases “root Folder” (project name, created by TestLink – “Sandbox” in Figure 2) must not be declared in Excel.
- The “Test Case” section (cyan columns header) can be used to declare Test Cases in TestLink (Test Cases are named “Test Case Operations” in TestLink GUI)
- The “Attachments” section (light green columns header) is used to list the number of already linked attachments.
- The “Custom Fields” section (dark pink columns header) is hidden by default. This section can be used to declare several Custom Fields in TestLink, by adding new columns if needed
- The “Steps” section (yellow columns header) can be used to declare Test Cases Steps and expected results in TestLink
- The “Coverage” section (grey columns header) can be used to declare Test Case coverage in TestLink
- The “Time Management” section (Orange columns header) is hidden by default. This section can be used to declare the Estimated execution Duration of a Test case
- The “Test Suite” section and the association of the “Test Case”, “Custom Fields”, “Steps”, “Coverage” and “Time Management” sections **must not** be filled on the same line

- All “Test Case” declared after a “Test Suite” section will be inserted in the most recent “Test Suite” defined
- “Test case”, “Custom Fields”, “Coverage” and “Time Management” sections of each Test case **must** be provided only once, even if the Test case is multi-steps
- Consecutives lines in the “Test Suite” section are considered as a “subfolders” (tree) declaration.
- The “Test Suite” **must provide the path from the root folder** each time a Test Suite is declared in the file. Test Suite “Name” and “Details” **must be** declared several times in the same file if the same folder is used for several sub-folders (see “Parent” example in “Figure 1” and “Figure 2”)
- Attachment can be added to TestSuites and/or TestCases. To add a new attachment, just use the hotkey “Ctrl+j” and select the file to join. The file is added to the TestSuite or TestCase highlighted when using the hotkey.

## Excel columns content explanations

(Red = mandatory, Black = Optional)

- **Columns used for a Test Suite declaration**
  - **Name**: TestSuite Name
  - **Details**: TestSuite Description
- **Columns used for a Test case declaration**
  - **TC#**: Test case Number. If filled, this number is used during import to determine the node order of the test in the TestSuite. If not filled, automatically generated by TestLink during import.
  - **Name**: Test case name
  - **Summary**: Test case short description
  - **PreConditions**: Test case Preconditions
  - **Status**: Test case Status. If not filled, automatically set to « Draft » by TestLink during import. Has to be filled with one of the following value :

Status in TestLink	Draft	Ready For Review	Review In Progress	Rework	Obsolete	Future	Final
Value to use in Excel	1	2	3	4	5	6	7

- **ExecutionType**: Test case Execution type

Value in TestLink	Manual	Automatic
Value to use in Excel	1	2

- **Importance**: Test case Importance

<b>Value in TestLink</b>	Low	Medium	High
<b>Value to use in Excel</b>	1	2	3

- **Keywords:** Keyword(s) that is/are linked to this Test case. To list more than one keyword for a test case, separator is semi-comma character (;)
- Columns related to attachments
  - **Number of Attachments:** Inform on how many attachments are known for the current TestSuite/TestCase. The number is automatically updated during XML Import and during the addition of a new attachment using “Ctrl + j” Hotkey.
- **Columns used for Custom Fields declaration**
  - User defined fields, according to Project definition
  - **Column title:** Custom field Name
  - **Cell content:** Custom field Value for each test case
- **Columns used for Steps declaration**
  - **Step:** What to do (action). One Excel row per step
  - **Expected Result:** Expected result for each step. One Excel row per expected result.
  - **StepExecutionType:** Step Execution type. Optional in Excel file, Mandatory in XML generated file. Forced to 1 (Manual) in XML file if not filled in Excel file.

<b>Value in TestLink</b>	Manual	Automatic
<b>Value to use in Excel</b>	1	2

- **Columns used for Requirements Coverage declaration**
  - **Spec Title:** Requirement Title that is covered by this Test case.
  - **Document ID:** Requirement Doc-ID(s) that is/are covered by this Test case. To cover more than one Doc-ID, separator is semi-comma character (;)
- **Columns used for Time Management declaration**
  - **Estimated exec. Duration:** The estimated duration to execute the test. In Minutes.

## Excel text length validation

Some TestLink fields are length limited. Each limit is configured in Excel file to inform the user when the limit is reached. When the limit is reached, the XML generation still works, but TestLink will truncate the value during import.

Each cell that contains too many characters is displayed with a red background.

- TestSuite Name: 75 characters maximum
- TestCase Name: 100 characters maximum

Test Suite		Test Case		
TestCases Generation				
Name	Details	Name	Summary	PreConditions
Parent	Details about the parent test suite			
myTestSuite where the name is very very very very very very long so the limit is reached	Details about my TestSuite			
		A first TestCase	The summary of the First TestCase	
		A second TestCase where the name is very very very very very very very very very very long so the limit is reached	A second test Case for My TestSuite	A precondition is defined
Parent	Details about the parent test suite			
YourTestSuite	Details about your TestSuite			
		Your first TestCase	The summary of the First TestCase of your testSuite	Your precondition
		Your second TestCase	A second test Case for Your TestSuite	

Figure 3 : Excel text length validation

## XML file generation

When all Test cases have been written in Excel, just click on 'Export Test cases To TestLink' button to execute the Excel Macro. Of course, macro execution must be allowed for the '02-TestCases.xls' file.

If there is a problem with the content, the popup should explain why the generation has failed.

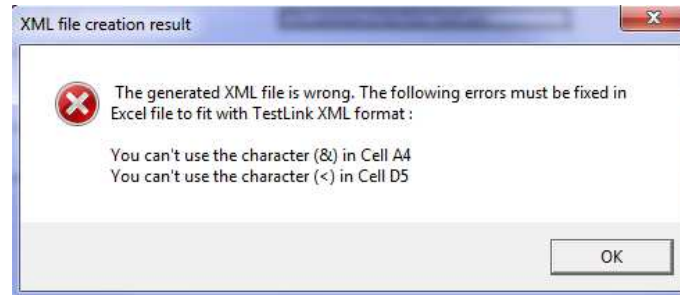


Figure 4 : XML file generation result KO

A log file containing all errors can be created if needed.

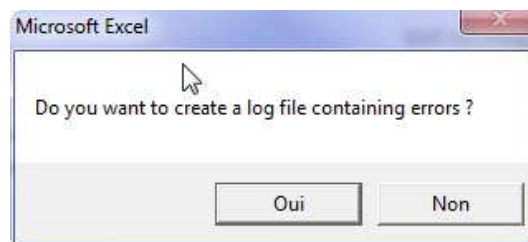


Figure 5 : Save the log file

The log file is created in the same Excel file folder, with the Excel file name (with a .log suffix).

If everything is OK with the file content, a popup will appear with the following text displayed



Figure 6 : XML file generation result OK

The XML file is created in the same Excel file folder, with the Excel file name (the .xls suffix is replaced by a .xml suffix).

When the XML generation is done, a popup ask if the requirement coverage must be done

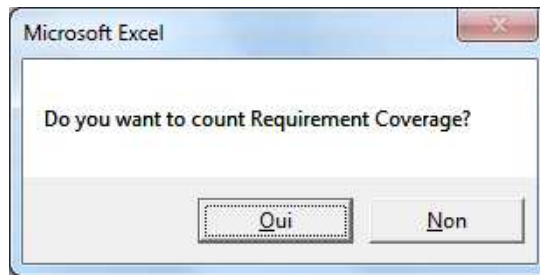


Figure 7 : Requirement Coverage question

If chosen, a new sheet is displayed, showing how many occurrence of each requirement is present

Requirements List	Requirements without duplicates	Occurrences
Req-SandBox-1	Req-SandBox-1	1
Req-SandBox-3	Req-SandBox-3	1
Req-SandBox-4	Req-SandBox-4	2
Req-SandBox-4		
Req-SandBox-2		

Figure 8 : Requirement Coverage result

This information should be useful to update the file "01-Requirements.xls" which has to provide the number of test cases needed to cover the requirement.

## Test cases Import into TestLink

XML file generated by the macro must be used during the import process.

- 1) Select the folder representing the root folder where all “Test Suites” and “Test cases” will be inserted. This can be the root folder or a folder in the Test Suites tree.

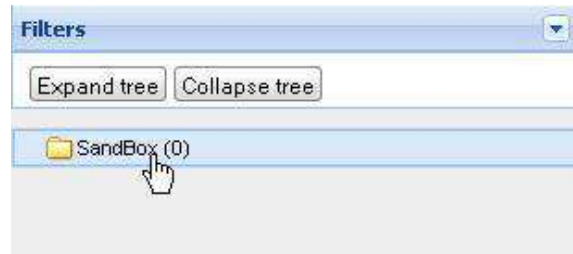


Figure 9 : select destination folder for requirements import

- 2) Select the import button

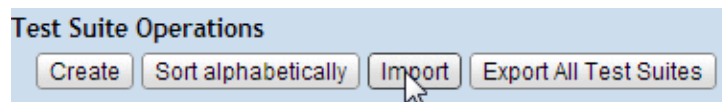


Figure 10 : Select import menu when the root folder is selected

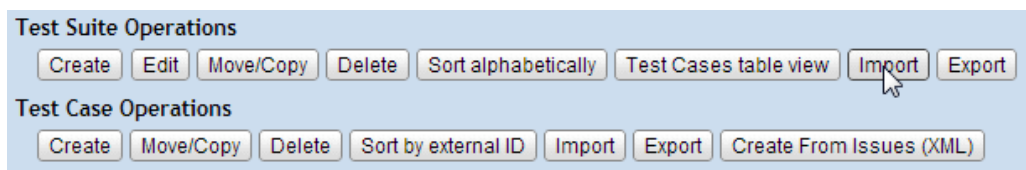


Figure 11 : Select import menu when a Test Suite (sub-folder) is selected

- 3) Configure the import options

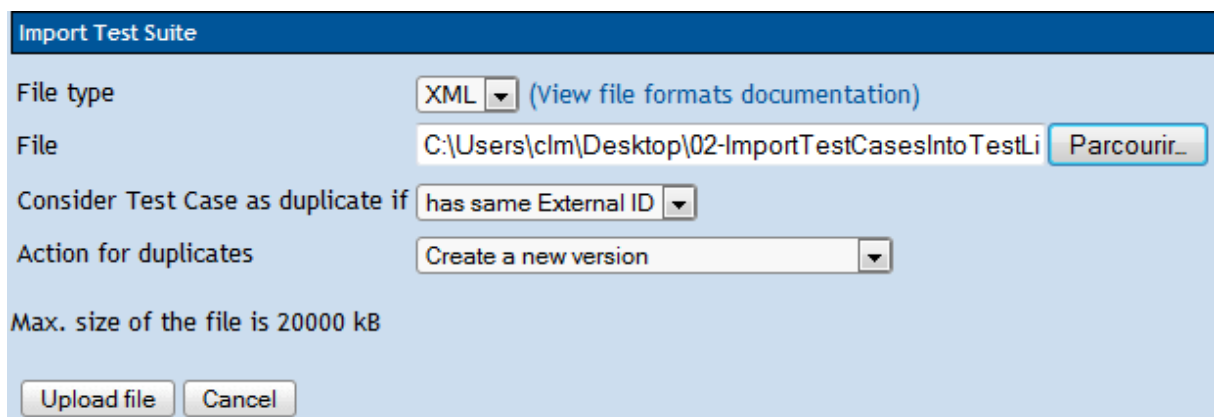
A screenshot of the 'Import Test Suite' dialog box. It has a title bar 'Import Test Suite'. Below the title bar are several fields and buttons. The 'File type' field has a dropdown menu set to 'XML' and a link '(View file formats documentation)'. The 'File' field has a text input 'C:\Users\clm\Desktop\02-ImportTestCasesIntoTestLi' and a 'Parcourir...' button. The 'Consider Test Case as duplicate if' field has a dropdown menu set to 'has same External ID'. The 'Action for duplicates' field has a dropdown menu set to 'Create a new version'. Below these fields is a text label 'Max. size of the file is 20000 kB'. At the bottom are two buttons: 'Upload file' and 'Cancel'.

Figure 12 : Test cases import options



Import Test Suite
Test Case : A first TestCase : ok
Test Case : A second TestCase : ok
Test Case : Your first 'TestCase : ok
Test Case : Your second TestCase : ok

Figure 13 : import Test cases result

## Import into Excel TestCases Exported from TestLink

To export from TestLink, use the Export Function provided by TestLink :

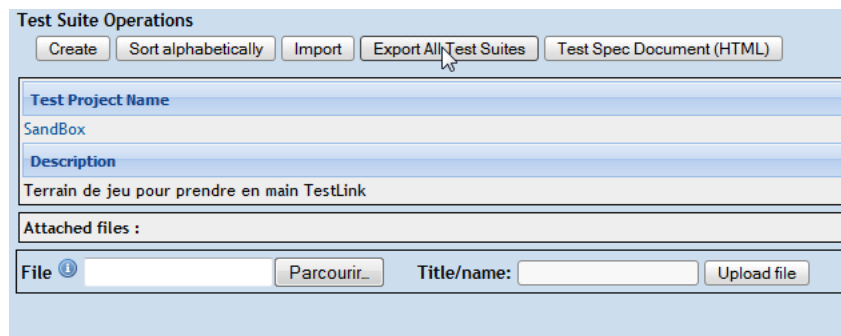


Figure 14 : Test cases Export function

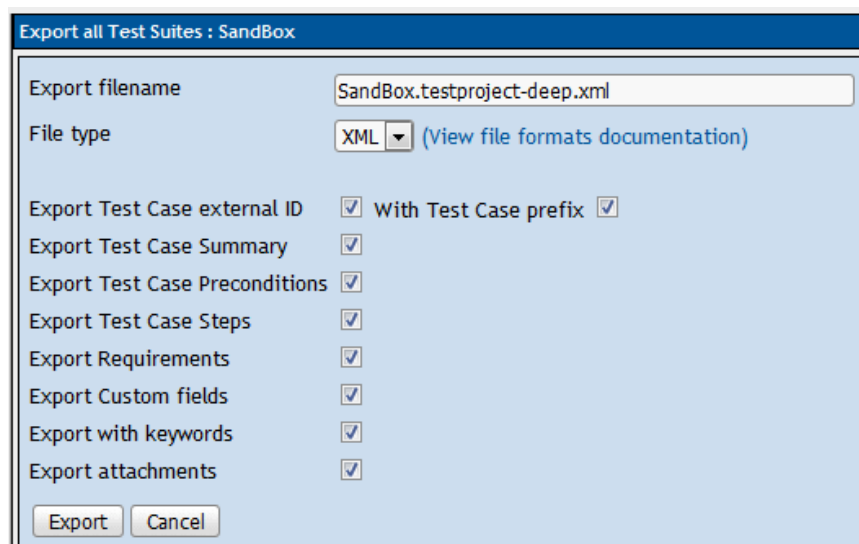


Figure 15 : Test cases Export options

### **WARNING :**

Test case content is depending of the text editor that you use when you write Test cases (CKEditor, FCKEditor, ...). The provided function should be used to import test cases **ONLY WHEN NO ADVANCED EDITOR IS USED** :

```
$tlCfg->gui->text_editor['design'] = array( 'type' => 'none');  
$tlCfg->gui->text_editor['steps_design'] = array( 'type' => 'none');
```

Editor internal information (HTML tags) will be showed if an advanced editor has been used, and Test cases content could be altered when importing back to TestLink.

If you want to import Test cases exported from TestLink, just click on 'Import Test cases From TestLink' button to execute the Excel Macro. Of course, macro execution must be allowed for the '02-TestCases.xls' file.

Test Suite		Test Case			
Name	Details	Name	Summary	PreConditions	Importance
Parent myTestSuite	Details about the parent test suite	A first TestCase	The summary of the First TestCase		1
	Details about my TestSuite	A second TestCase	A second test Case for My TestSuite	A precondition is defined	2
Parent YourTestSuite	Details about the parent test suite	Your first TestCase	The summary of the First TestCase of your testSuite	Your precondition	3
	Details about your TestSuite	Your second TestCase	A second test Case for Your TestSuite		2

Figure 16 : Test cases Import button in Excel

A confirmation message is displayed; the user must acknowledge that all current data in the spreadsheet will be lost.

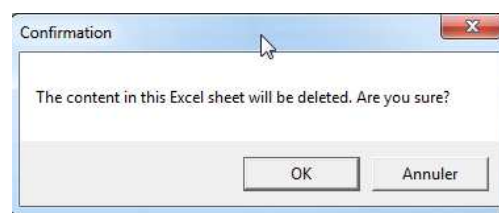


Figure 17 : Confirmation message

The file generated by TestLink during export must be provided.

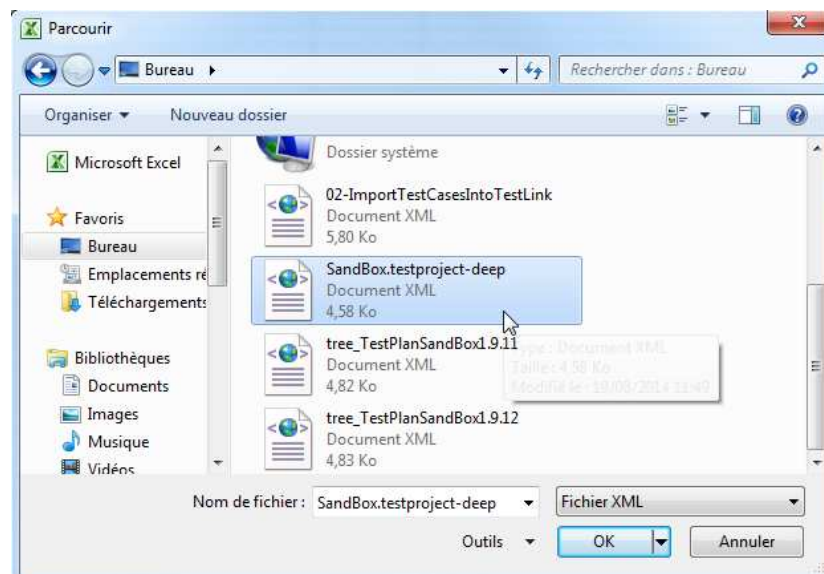


Figure 18 : XML File Selection

The file generated by TestLink is parsed by the Excel macro. Depending of the content, this analyze can be time-consuming. A GUI inform user about import progression.

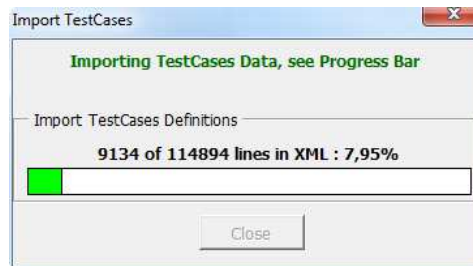


Figure 19 : XML Import Progression GUI

When finished, the Excel content represents the content of each test case.

Test Suite		Test Case				Steps		Coverage
Name	Details	Name	Summary	PreConditions	Importance	Step	Expected Result	Document ID
Parent myTestSuite	Details about the parent test suite Details about my TestSuite	A first TestCase	The summary of the First TestCase			1 The first Step A second Step	Expected result for the first step And an expected result for the second step	
		A second TestCase	A second Case for My TestSuite	A precondition is defined		2 The first Step A second Step	Expected result for the first step And an expected result for the second step	
Parent YourTestSuite	Details about the parent test suite Details about your TestSuite	Your first TestCase	The summary of the First TestCase of your testSuite	Your precondition		3 The first Step A second Step	Expected result for the first step And an expected result for the second step	
		Your second TestCase	A second test Case for Your TestSuite			2 The first Step A second Step A third Step	Expected result for the first step And an expected result for the second step Expected result for the third step	

Figure 20 : Result after Import into Excel

If you want to edit “advanced” information about test cases (custom fields, execution type, External ID, Keywords), you have to display hidden columns.

You can now edit test cases information and import content back to TestLink to create a new version or update the last version.

After an export, the External ID is correct in the Excel file, the best configuration to Import back into testlink is to use External ID as identifier. With this provided configuration, you can modify the Test case name if needed :

**Import Test Suite**

File type
XML (View file formats documentation)

File
C:\Users\clm\Desktop\02-ImportTestCasesIntoTestLi
Parcourir...

Consider Test Case as duplicate if
has same External ID

Action for duplicates
Create a new version

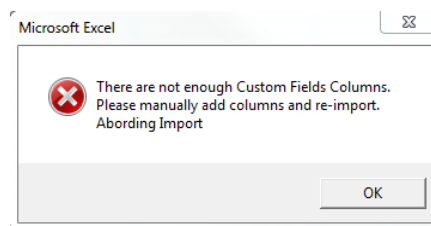
Max. size of the file is 20000 kB

Upload file
Cancel

Figure 21 : Import Test cases update – preconized configuration

## Tips and problems

- By default, **columns representing optional information** are hidden in Excel file (External ID, Test case Execution Type, Keywords, Custom Fields, step Execution Type and specification title for coverage)
- In **Coverage Document ID** and **Keywords** columns, if more than one value is provided, each value can be provided on its own row (Alt + Enter to add a Carriage Return in the cell), separator between each value is semi-comma (;)
- During import, if XML file presents more **Custom Fields** values that Custom Fields Columns in excel, import is cancelled to let user add more columns.



- In **Custom Fields columns**, if more than one value has to be provided (Multi Selection List type), each value has to be separated by | character.
- During **keywords** import, already linked keywords cannot be unlinked. The import only add new keywords to test cases.
- **Generate a Test cases XML file** with Excel has been tested with Excel 2003, Excel 2007, 2010 and 2013 versions.
- **XML Test Suite** haven't to be created in TestLink before import
- If your **Test cases titles** are longer than 100 characters, titles will be reduced by TestLink during import. You have to modify const.inc.php to reconsider the value of test case title field size (max is 255 caracters in database)
- If your **XML Test cases file** is bigger than 400Kb, you have to edit custom\_cfg.inc.php to change "\$tICfg->import\_file\_max\_size\_bytes" parameter. To set new limit file size to 1Mb, modify this parameter from '409600' to '1024000'.
- The **XMLTest cases file encoding** depends of Excel version. In French version, the Excel generated file is not UTF-8 but ANSI. Encoding information is set to "ISO-8859-1" in XML file, allowing the use of Latin characters (like é, è or à used in French). If a "XML Loading Failed" is displayed during XML import in TestLink, check the file encoding.
- **Attachments** can be linked to TestSuites and TestCases only. TestLink cannot store attachments related to Test Steps. This 02-TestCases.xls file doesn't provide attachments for steps.

- **Attachments** can be duplicated after an export from TestLink + import to TestLink if you don't upgrade your TestLink version to TestLink 1.9.18.
- **Attachments** are stored on the File System in the same folder than 02-TestCases.xls file. All attachments are stored in "TestSuites" and "TestCases" folders, depending of the linked item. Each "TestSuite" and "TestCase" with attachment has his own folder to store attachments, using "TestSuite Name" or "TestCase-External-ID" value as identifier.
- **Attachments** storages are based on "TestSuite Name" and "TestCase-External-ID" values. If you rename one of these values, you'll lost your linked attachments except if you rename the File System folder accordingly.
- New TestCases can be linked to **Attachments**. The storage is done using a temporary external-ID starting with pattern "TMP-". You can view the auto-generated value in hidden column "External ID". Test ignores the "TMP-" value during import because the format is not compliant with the expected numeric format. Final External ID is generated by TestLink during import, attachments are preserved.
- **Attachments** related to "root folder" (project name, created by TestLink) are not handled by this file. These Attachments cannot be imported/exported.
- During Import from XML file, **Attachments IDs** are concatenated to attachments names because IDs are needed during import back to avoid duplicates in TestLink. The chosen format is #ID#AttachmentName. If the ID isn't set or if it is different than the already known in database, TestLink will create a new attachment during XML import.
- In **Number of Attachments column**, the number is based on real number of attachments copied to File System in the right directory, not only an increment in the Excel column. This is done to avoid mistakes introduced by duplicates attachments.
- In **Number of Attachments column**, the value is informational, Export to XML will use all known attachments even if this number is wrong. You can manually copy/paste attachments to a TestSuite or a TestCase in dedicated folder if needed.
- During **import into TestLink**, TestLink doesn't accept that two test cases have the same name in the same TestSuite.
- In **import result log view**, you can search for "ok", "skipped", "updated" or "duplicate" keywords to verify that TestLink Test cases import comportment is coherent with your needs.
- In **import from TestLink**, node hierarchy deep is configured with a maximum of 100 "sub-nodes". If value must be adapted, variable tTsInfo in function "ImportTestCasesXML" () should be modified.

# Document Changelog

Excel File Version	Major Changes
20120314	Initial Release
20130218	Explain the Requirement Coverage function Import has been successfully tested within TestLink 1.9.3, 1.9.4, 1.9.5 and 1.9.6
20130530	Explain the Requirement Coverage function Import has been successfully tested within TestLink 1.9.7
20130916	Update How To information
20140129	Update How To information
20140403	Update How To information to explain the use of sub-folders tree solution (multi TestSuites declaration)
20140415	Update How To information to explain how Excel is used to validate text length  Import has been successfully tested within TestLink 1.9.8 and 1.9.9  Add explanations about the new XML tag <estimated_execution_duration>
20140724	Update How To information.  Import has been successfully tested within TestLink 1.9.10

Excel File Version	Major Changes
20140827	Update How To information to explain how to use the "Import Test cases From TestLink" button.  Import has been successfully tested within TestLink 1.9.11
20141214	Update How To information to fit macro comportment.  Import has been successfully tested within TestLink 1.9.12
20150331	Bug correction : Max sub-node hierarchy is now configured to 10 (complex node hierarchy).  Import has been successfully tested within TestLink 1.9.13
20150831	Bug correction : first requirement is displayed twice in requirement counter
20160209	Add information about custom fields format regarding custom field type  Update package version information
20160229	Add information about Customs Fields and Keywords exported from TestLink  Update package version information  Import has been successfully tested within TestLink 1.9.14
20160311	Add Custom Fields Import information  Add Test cases Status information  Add Keywords format description, and information about import/export of keywords  Update package version information
20160331	Add XML Import Progression GUI section  Update package version information
20160731	Update package version information



Excel File Version	Major Changes
20170501	Update information based on version 20170424  Tested with Excel 2013 and with TestLink 1.9.17-dev capacities
20180514	Add attachments explanation.  Warning: TestLink 1.9.18 version is mandatory to import attachments without generate duplicates.