SEFELEC REFERENCE PENT3882

E·AT·N

Powering Business Worldwide

Table of contents / Table des matières / Inhaltsverzeichnisses

Chapter
Chapitre
Kapitel

I

Conventions

I - Conventions

Instructions for using the software version 1.16

The following paragraphs explain some key words used in this document.

Install directories

[installDir] : Default install directories for Winpass 5000 (C:\Sefelec\Winpass5000).

[AccountDir] : User accounts default install directories, for Winpass 5000 access (C:\Sefelec\Winpass5000\accounts).

[calibrationDir] : répertoire d'installation par défaut des fichiers de calibration du testeur (C:\Sefelec\Winpass5000\calibration).

[connectorDir] : Winpass 5000 Default install directories for connector library (C:\Sefelec\Winpass5000\connectors).

[DllDir] : répertoire d'installation par défaut des DLL utilisées par le logiciel (C:\Sefelec\Winpass5000\DLL).

[ErrorDir] : répertoire d'installation par défaut des enregistrements des erreurs rencontrées par le logiciel (C:\Sefelec\Winpass5000\Error).

[ExecutionConfigDir] : répertoire d'installation par défaut des configurations d'exécution du logiciel (C:\Sefelec\Winpass5000\ExecutionConfig).

[FirmwareDir] : répertoire d'installation par défaut des versions de firmwares installées sur les cartes équipant le Testeur (C:\Sefelec\Winpass5000\Firmwares).

[InterfaceDir] : répertoire d'installation par défaut des interfaces de raccordement équipant le Testeur (C:\Sefelec\Winpass5000\interfaces).

[ProjectsDir] : répertoire d'installation par défaut projets de test (C:\Sefelec\Winpass5000\projects).

[TesterDir] : répertoire d'installation par défaut des configurations de testeurs (C:\Sefelec\Winpass5000\testers).

[editorDir] : Winpass 5000 Default install directories for the editor tools (C:\Sefelec\Winpass5000\editor).

[macroDir] : Winpass 5000 Default install directories for the macros (C:\Sefelec\Winpass5000\macros).

[testerDir] : Winpass 5000 Default install directories for the various cable tester setups (C:\Sefelec\Winpass5000\testers).

Examples

[exDir] : Winpass 5000 Default install directories for the provided examples from the software installation (C:\Sefelec\Winpass5000\projects\demo).

Notes

Notes and important points of that document are displayed in grey areas.

Chapter
Chapitre
Kapitel

II
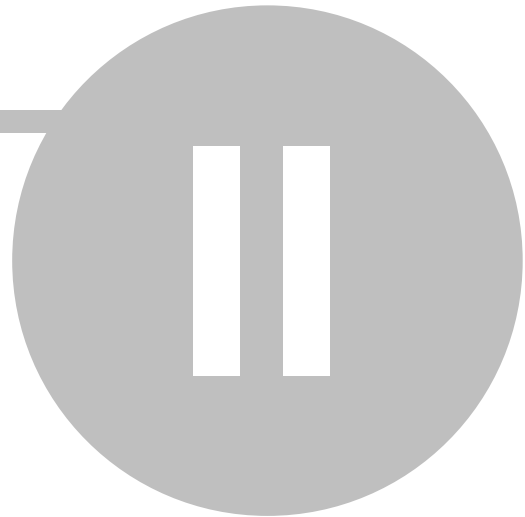
Installation

II - Installation

Testers are delivered with a CD ROM containing the WINPASS software, options and tester configuration.

Simply insert the CD ROM in the drive and the program loads automatically. The installation software lets you choose the work directories. To accept the default directories, click on next in each prompt.

⚠️

NB: If the testers are in network, you must be in administrator mode and the operation files must be on the PC disk.

1 - Installing Winpass5000
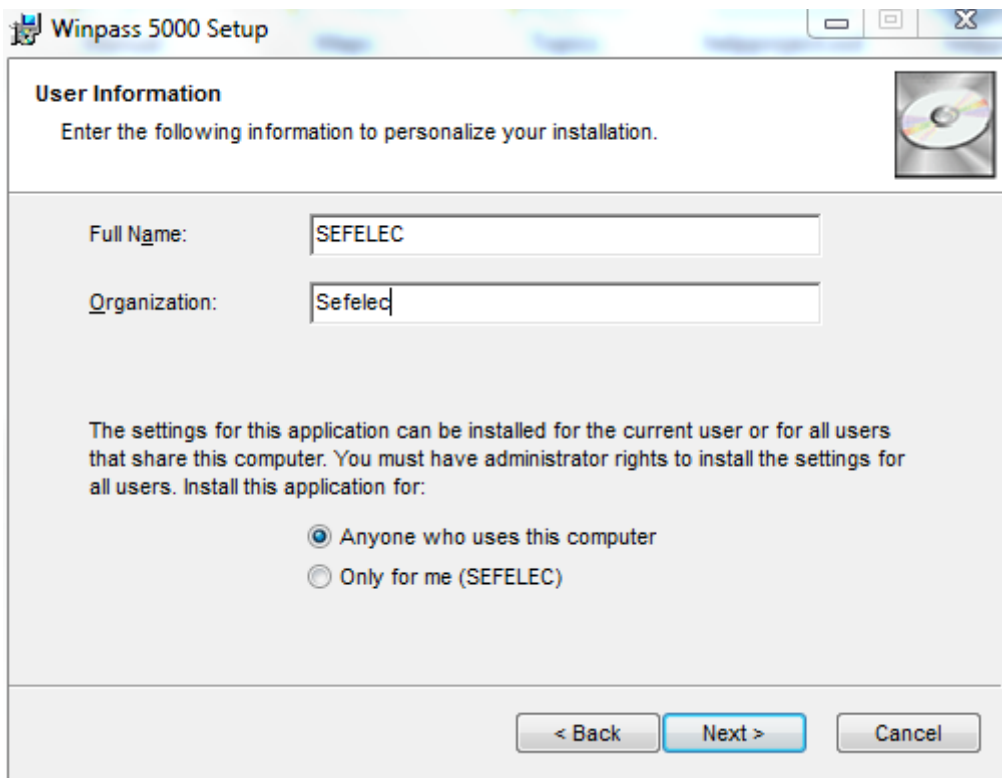
Installation description :

- Put the CD-ROM in the PC drive

Installation starts automatically and the following window appears.

- Click Next.

- Give the name and organisation and click Next.

By default, the program is installed in C:\SEFELEC\Winpass5000.
- Change the installation directory, if required, then click Next.

Installation is automatic up to the completion window.
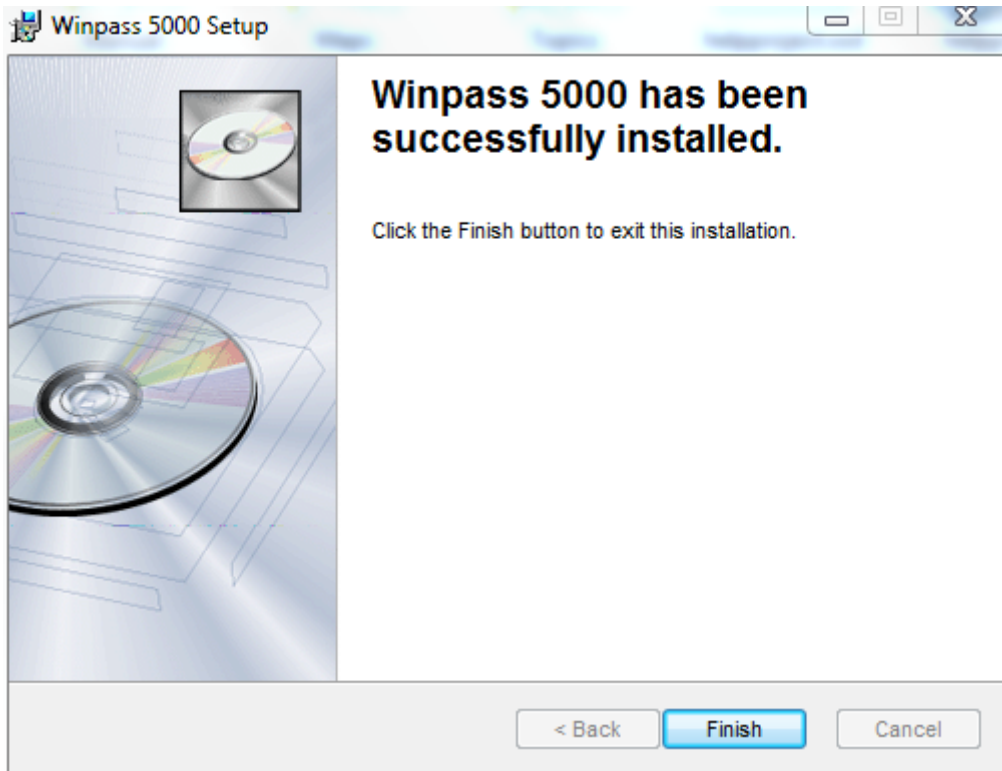
- Click Finish to finish the installation.

The Winpass icon should now be on the desktop.

WinPass5000 is now installed.

2 - Setting up the Computer's Ethernet Card under Windows XP

- Go to Start / Control panel / Network and Internet connections
- Right click the Ethernet card connection and click Properties
- Select the General tab
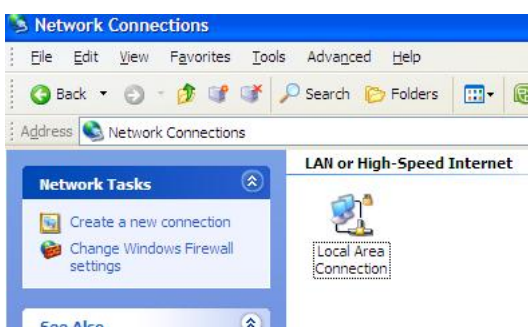
- Select Internet protocol (TCP/IP) in the field "This connection uses the following items" and click Properties
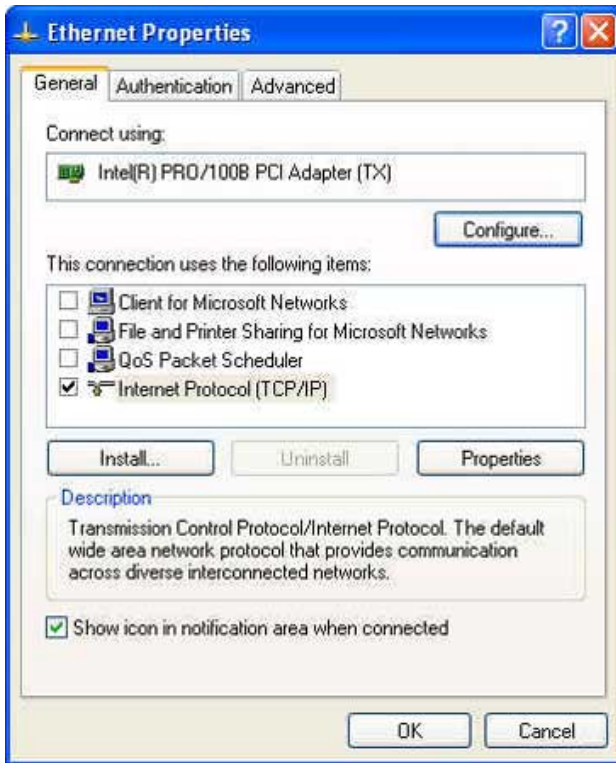
- Select Use the following IP address

- Set the IP address of the PC network card (Eg 192.168.64.100)

- Set your IP subnet mask (Eg 255.255.252.0)
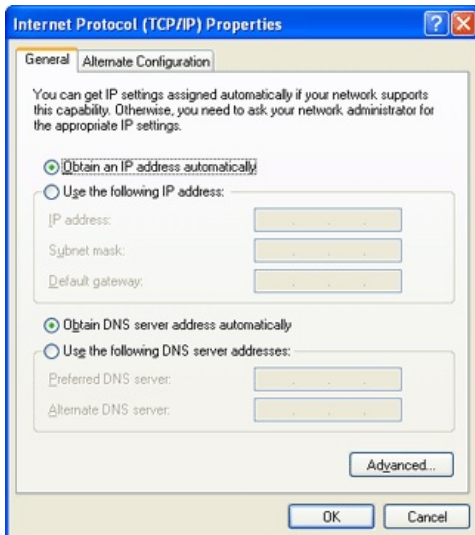
- Click OK to close the Properties window. Click OK to close the Network connections window.

3 - Setting up the Computer's Ethernet Card under Windows 7

- Go to Start / Control panel / Network and Internet
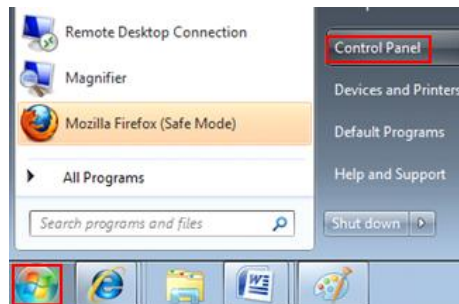
- Click View network status and tasks

- Click Change adapter settings
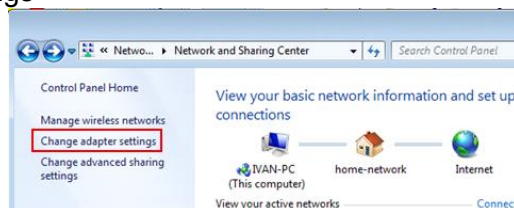
- Right click the Ethernet card connection and click Properties
- Select Internet protocol version 4(TCP/IPv4)
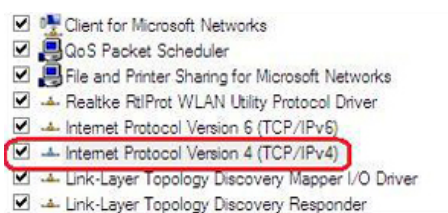
- Click Properties
- Select Use the following IP address

- Set the IP address of the PC network card (Eg 192.168.64.100)

- Set your IP subnet mask (Eg 255.255.252.0)
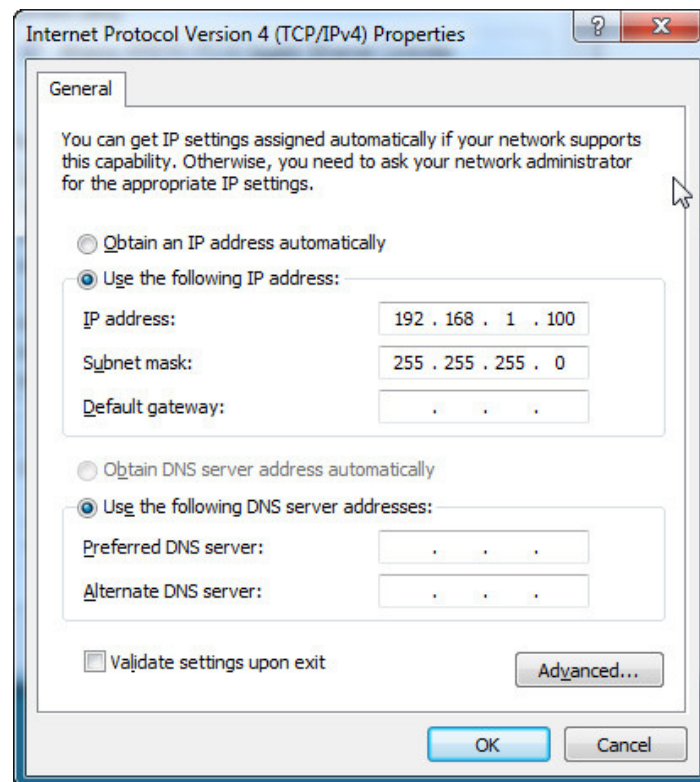
Click OK to close the Properties window. Click OK to close the Network connections window

Chapter
Chapitre
Kapitel

III

Interface

III - Interface

Before been able to use Winpass 5000, it is convenient to understand its operating principle and organization.

Winpass is a software that serves as an interface between an operator and an industrial system; which is in this case a Synor 5000 cable tester. Therefor, it needs to deliver every tool permitting **the use, management and verification of the Synor 5000**.

On the other hand, it is dedicated to execute electrical tests of a product while in its manufacturing cycle or while in maintenance or repair. In this optic, Winpass allows **conceiving or producing tests needed to validate a product**.

Furthermore, it allows achieving electrical integrity tests (cable tests) but also functional tests (Product behavior while powered up).

As a result, Winpass revolves around three main modules :

- A **"Tester" module** dedicated to communicating and managing a Synor 5000 properties/options.

- An **"Edit" module** dedicated to conceiving test projects on a specific product.

- An **"Execute" module** dedicated to undertaking tests and producing tests reports

We will see in this document that Winpass offers a wide variety of specific tools enabling you to broaden and facilitate your work, but also complementary functions dedicated to validate projects or specific products.

It is also mandatory to understand that **Winpass is "Project" oriented**. This mechanic provides the possibility of combining, at your convenience, every mandatory test needed to validate a product, a products group or a product composed of several subsets.

Given that Winpass is very rich in term of tests functions, it has been designed both to give you essential information at the right time, and give you productivity, design & test tracking tools.

The Winpass 5000's interface breaks down in **work zones (or panels)** that allow identifying clearly points that we have just described. Just like every software and in order to perform its tasks, Winpass possesses menus that, in an effort to keep it clear and simple, are shown as **ribbons**. Its other contribution is to provide quick access to information and handled files, and that is why it is organized around **tabs**.

The following figure is a comprehensive representation of every aspect we have described until now :
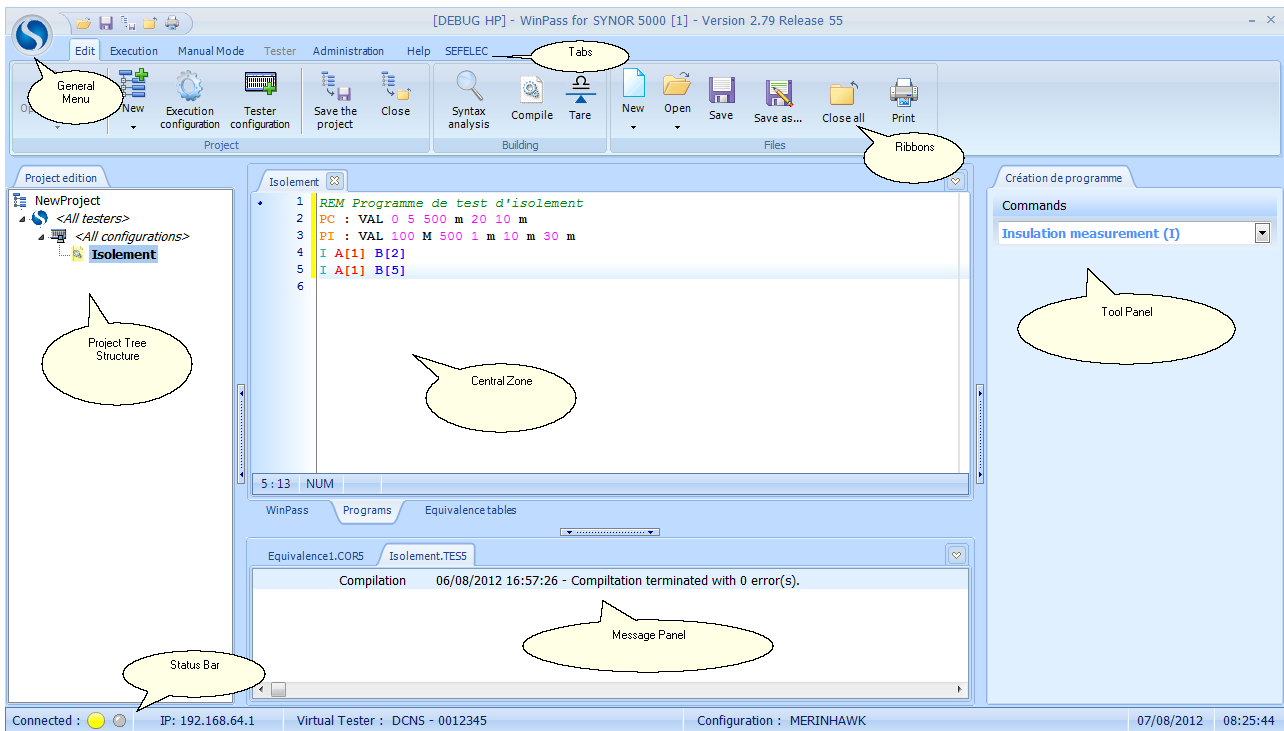
Winpass 5000 interface is in line with Windows 7' standards, such as the Office suite, therefore we can find common elements that can prove familiar :

- A **general menu** (and its quick-access bar) dedicated to managing Winpass files.

- **Tabs** used for accessing Winpass modules and merging their functions.

- A **ribbon** displaying a specific module's dedicated functions.

- A **status bar** showing information about the connection with a tester.

- A **left panel** devoted to display and manage projects

- A **right panel**, dedicated to design tools but also execution tracking.

- A **work zone (central zone)** regrouping every file currently edited or executed.

- A **lower panel** gathering messages or events links to every file in progress.

IMPORTANT :

Depending on which modules you use or tasks currently executed, some of these elements can be either hidden or inactive.

In its current shape, and even in Winpass' future iterations, this organisation will never change; however, different elements composing it could be upgraded with new functions.

---

1 - General Menu

The general menu can be accessed by left-clicking the Winpass 5000 logo :

IMPORTANT

This menu is only accessible (active) when Winpass is in edition mode, meaning whenever the "Edit" module (or tab) is active.

This menu allows you to create, load, save and print files in edition. The right panel lists the 10 last saved files or types of editable files ("New" and "Open").

Recent files

Demo.PRJ
Location: C:\SEFELEC\Winpass5000\Projects\Demo\

Demo_Distribution.PRJ
Location: C:\SEFELEC\Winpass5000\Projects\Demo_Distribution\

Test_Renault2.CDF
Location: C:\SEFELEC\Winpass5000\WinTransfert\Distributions\

Demo_SlpicePoint.xls
Location: C:\SEFELEC\Winpass5000\WinTransfert\Excel\

ADA_HAL.xls
Location: C:\SEFELEC\Winpass5000\WinTransfert\Excel\

Demo_Distribution.xls
Location: C:\SEFELEC\Winpass5000\WinTransfert\Excel\

Test_Serie_Cable.WFT
Location: C:\SEFELEC\Winpass5000\WinTransfert\Models\
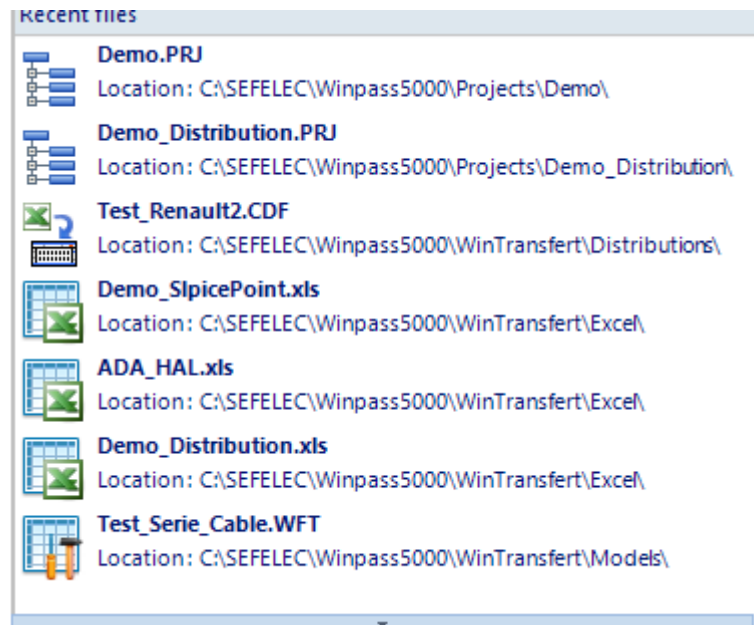
IMPORTANT :

Concerning the "Save", "Save as", "Close" and "Print" functions, they applies to the file currently being edited (meaning the active file in the work zone).
They are of course deactivated while no file is currently edited.

The general menu is also composed of a quick-access bar, regrouping the most frequently used functions and working the same way.

New   ▶

Open   ▶

Save

Save as...

Save the project

Close

Close all

Print
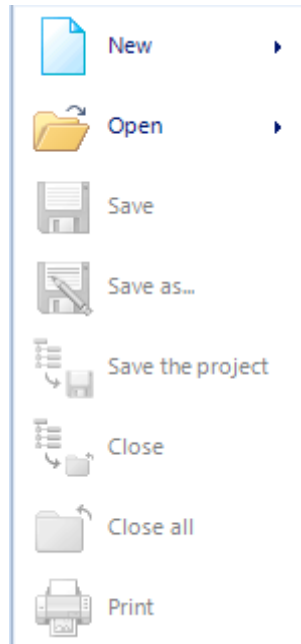
2 - Tabs & Ribbons

Tabs allow you to activate different Winpass 5000 modules. Once you have selected a module (left click on a tab), you gain access to this module's tasks by displaying its functions in the ribbon.

A module's functions (buttons, submenus, checkboxes, prompt boxes, ...) are clustered by affinity (Projects, Programs, Files, Tools, ...) in a clarity concern.

IMPORTANT :

Activating a module or selecting a function directly affect Winpass behavior. Therefore, some modules are only visible or usable under certain conditions (For instance, the "Execute" tab is only usable if you are connected to a tester, and the "Tester" tab is unattainable while a project is currently being executed).

The same goes for a ribbon's function of a specific module, or while a task is ongoing, it can display, hide, allow or forbid particular functions of other modules.

Note :

We are hereby describing standard Winpass tabs. We will see, where required, that other tabs and functions linked to specific Winpass or testers options exist (WinPV, WinTransfert, Standalone mode, Barcode scanner, Simplified execution interface, …).

2.1 - Edit

The Edit tab will allow you to conceive projects, create and manage files linked to a product's test, but also to verify and check your files before using them.

> **CONVENTION :**
> A **tests file currently being edited** means that it is active and/or visible in Winpass' work zone.

In this module, functions are sorted in three groups :

1 -The "Project" group manages a project conception :
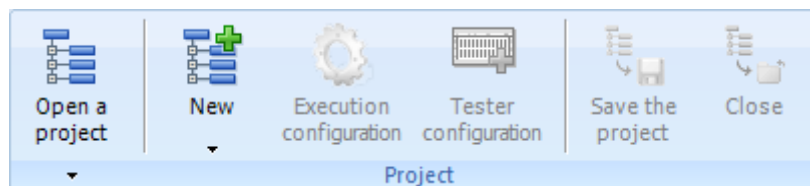
- "Open a project" : Lets you open a project in edition in order to modify it.

- "New" : Allows creating a new blank project but also linking new blanks test files to a project.

- "Execution configuration" : Permits adding or managing a project's execution options.

- "Tester configuration" : Provides a way to condition the execution of specific tests to a certain tester.

- "Save the project" : Lets you save on your hard drive any modification done to a project.

- "Close" : Allows closing the project currently being edited.

> **IMPORTANT :**
> When saving or closing a project, every test file (linked to this project) currently being edited will be automatically closed too.

> **Note :**
>
> When closing, for every new or modified project, and for every file (linked to the project) currently edited or modified, Winpass will display a dialog box asking you if you want to save the project & each document.

2 - The "Files" group is dedicated to file management relative to products tests :

- "New" : Permits creating a blank test file.

- "Open" : Permits opening an existing file in edition in order to modify it.

- "Save" : Permits saving the file in edition.

- "Save as …" : Permits saving a new file currently edited or saving a copy of the current file on the hard drive.

- "Close all" : Permits closing every opened file and possibly the current project (& de facto, every file linked to it).

- "Print" : Permits printing the current file on the default printer.

Note :

When saving or closing a test file, two instances can occur :

- If the file is new, Winpass will show a box asking you to enter the file name & choose the save directory.

- If the file already exists and is modified, Winpass will display a box asking you whether you want to save or not.

---

3 - The "Building" group is used to verify and validate test files :
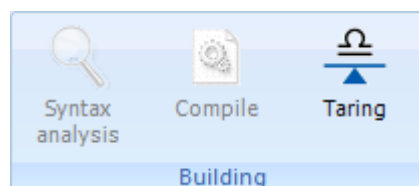
- "Syntax analysis" : Permits verifying that the current document follows the syntactic rules of its own programming language (depending on the file type).

- "Compile" : Permits translating a file's command lines to Winpass internal processing data, but also to check instructions compatibility among them.

- "Taring" : Permits handling tare weights that need to be subtracted for specific measures.

IMPORTANT :

When compiling a file, Winpass controls its compatibility with its associated tester or, by default, the current tester (connected to Winpass).

2.2 - Execute

The execute tab allows you to conceive your products tests by executing whole projects or specific test programs. It can also permits you to manage your products tests' results.

> **IMPORTANT :**
>
> This tab (or module) is only accessible while connected to a tester (either physical or virtual).

> **CONVENTION :**
>
> An **ongoing test program** means that it is active and/or visible in Winpass work zone.
>
> A **stopped** project/program means that it has been fully executed or prematurely stopped.

In this module, functions are sorted into four groups :

1 - The "Project" group handles projects execution :

- "Load" : Permits loading a project in order to execute it.

- "Execute" : Permits executing a test of one or several products by executing every program from the project one following the other.

- "Stop" : Permits stopping the execution by immediately putting an end to an ongoing test program.

- Automatic : Permits chaining a project test programs execution (from the first to the last) without interruption between each.

- Pause : Permits marking a pause between each program's execution; the operator has to start again manually the project execution in order to advance to the next program.

- "Close" : Permits closing an ongoing project.

> **IMPORTANT :**
>
> Closing an ongoing project does not erase its state; loading the project later on will restore the same saved state as in its last use, and executing it will continue from its last non-finished program, or the next one if it was completely executed.

> **Note :**
>
> The state of execution of a project's tests programs is saved when the project is closed.

The next time you will load the project in execution, its test programs state will be restored exactly as during its last execution.

2 - The "Tests' program" group manages tests programs execution individually :
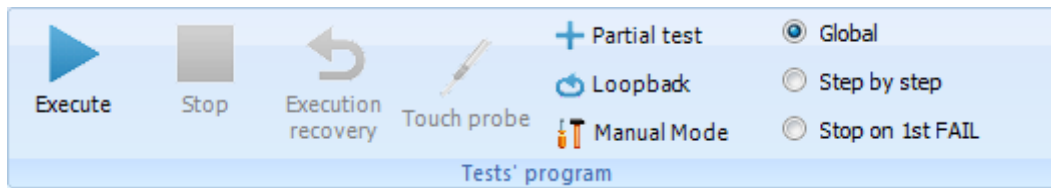
- ▶ "Execute"/"Continue" : Permits performing a test on one or several products by executing the current tests program.

- ⏸ "Pause" : Permits marking a pause during an ongoing program execution; Winpass waits for the last test instruction to be finished before pausing the execution.

- ⏹ "Stop" : Permits stopping sooner the ongoing test program's execution; Winpass does not wait for the last test instruction to be finished.

- ↩ "Execution recovery" : Permits accessing the loopback module of a product's test previously executed while only treating FAIL tests.

- ╱ "Touch probe" : Permits searching for electrical links with a touch probe connected to the tester.

- ➕ "Partial test" Option : Permits selecting specific parts of an ongoing program that should or should not be executed.

- 🔄 "Loopback" Option : Permits activating the automatic loopback of an ongoing program test' execution starting back at the beginning when it reaches the end.

- ⬆T "Manual Mode" Option : Permits activating the transition to manual mode whenever the program execution reaches a defective measure.

- ◉ Global : In this mode, test programs will be automatically executed in their totality.

- ○ Step by step : In this mode, program instructions will be executed one by one, marking a pause between each of them; the operator has to manually start again the execution for it to keep going.

- ○ Stop on 1st FAIL : In this mode, Winpass keeps executing the program while no test nor measure is defective; If this ever happens, execution is stopped and the product's test is ended as FAIL.

- ○ Pause on 1st FAIL : This mode is an variant of the previous one; in that case, the program's execution is put in pause on a FAIL test, and the test product is not closed (the operator can manually start the execution once again to check the product).

IMPORTANT :

Even is several tests programs can be opened in execution, Winpass only handles one execution at a time; This means that a tests program can only be executed once every other is finished (Totally executed, or the executions has been stopped early).

Note :

Options and modes apply to every tests programs opened in execution.

Apart from the "Partial Test" option, every other one are customizable while executing a tests program. However, they can be active/non-active or even banned depending on choices made in the current project's execution configuration.

---

3 - The "Tools" group permits executing several operations on the project's files :
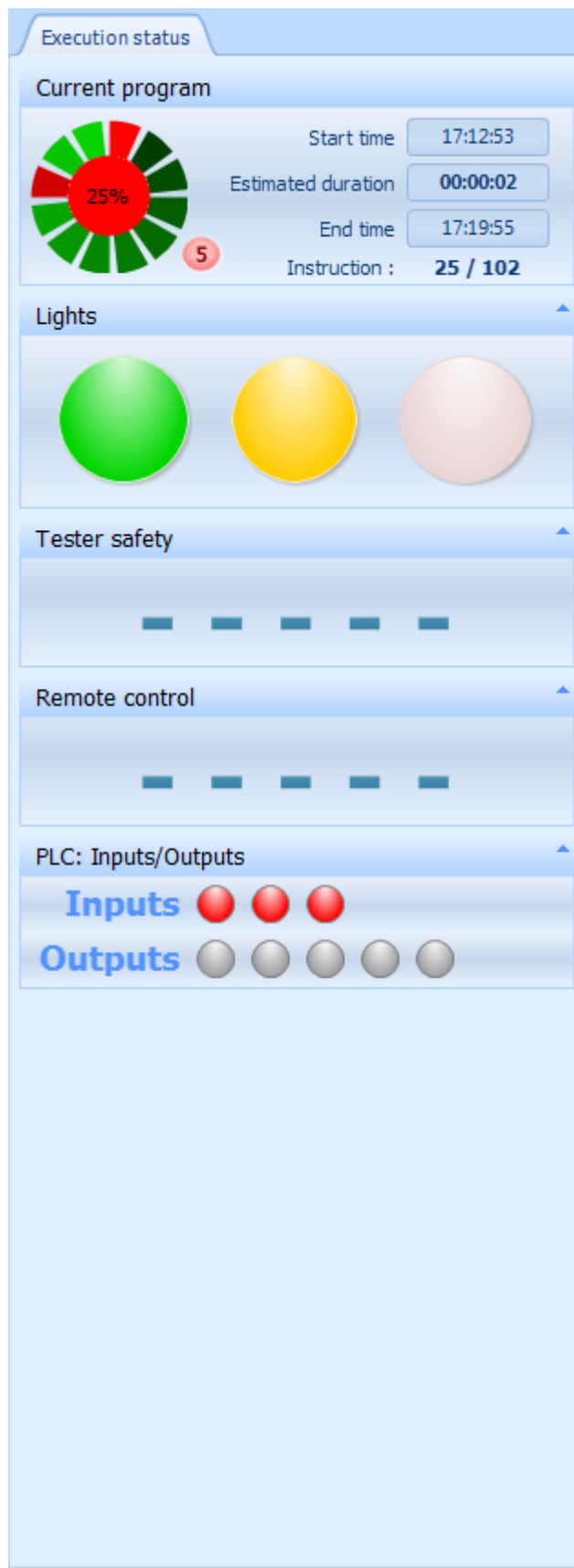
- : Permits executing the product's tests using a simplified execution interface; This module is also called "Operator Mode".

- "Load" : Permits choosing and loading the current test program's last results; This tools is particularly used to perform some "Execution Recoveries".

- "Save" : Permits saving the file(s) currently displayed in Winpass ork zone; This can either be test programs or tests result files (RTF and/or Excel).

- "Print" : Like the previous tool, this one permits printing files currently displayed (Program or tests results).

IMPORTANT :

The "Load" tool uses a file format that references the whole test program execution context, as well as every test results; This file format (RES5) is not handled by default, it has to be specifically activated in the project's execution configuration before any test program execution.

Note :

Saving and printing test results can be done even if the program's execution has not reached the end (Paused execution, for instance).

The "Execution interface" tool is only visible and usable if it has been specifically activated in Winpass' configuration; By default, it does not appear in the "Tools" group.

4 - The "Results management" group handles the results display modes :
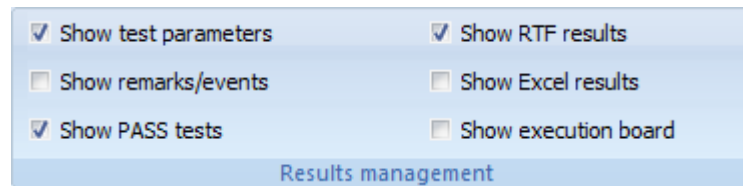
- : Permits choosing if tests and measurement parameters are displayed in results files (RTF & Excel).

- : Permits choosing if remarks and execution events are displayed in results files (RTF only).

- : Permits choosing if the PASS tests and measurements are displayed in results files (RTF & Excel); FAIL results are always displayed.

- : Permits displaying results in an RTF file.

- : Permits displaying results in an Excel file.

- : Permits displaying an execution tracking module, regrouping every treatment and tests during the execution of a program.

IMPORTANT :

Options and results display modes don't influence neither saving nor printing; this can be managed in the project execution configuration panel.

Whenever you modify an option or switch modes, Winpass automatically regenerate test results visible in the work zone; the regeneration time can vary depending on the number of messages and data available ofr the current execution result (This operation can take up to several tens of senconds for the biggest tests programs).

Note :

Changing results options and display modes only affects the currently visible tests program; Switching to a tests result to another will restore options and modes.

2.3 - Manual Mode

The manual mode tab will allow you to execute tests and measurements directly on your products without using a tests program. This mode allows you to check or validate a modification or reparation done on a product before or during its validating test.

IMPORTANT :

This tab (or module) is only accessible while connected to a tester (either physical or virtual) and when no tests program is being executed.

CONVENTION :

A **Measurement** is a process doing one or several surveys of electrical or electronic values, producing at least a physical value as a result (Ohm, Volt, Ampere, …)

A **Test** is a complex process doing a set of measurement sanctioned by a PASS or FAIL state.

Note :

Manual mode is not appropriate if you plan on doing a complete product's test, as it generates no measurement statement data or file to validate your product.

In this module, functions are sorted bewteen three groups :

1 - The "Tests and measures" group allow you to choose the test or measurement type to execute :

Continuity measurement

PC RMIN= 0 Ω   RMAX= 5 Ω

Tests and measures

- : Permits selecting and parametrize measures types; but also condition the validation or non-validation of tests (PASS or FAIL state and valid values).

- : Permits viewing the most important current test or measurement parameters (a test can be composed of several measurement types).

- : Permits accessing the test and measurement options panel dedicated to configuring the current tester. These options parametrize the context in which the tester will have to execute those measurements.
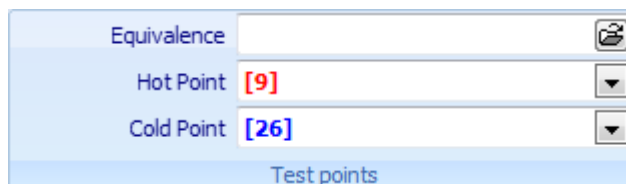
IMPORTANT :

Tests options from the Manual Mode are mutual to every tests and measurements.

Test or measurement type conditions if the tester handle or not these options.

Note :

The tests and measurements list depends on the current tester's options as well as external devices usable by it. Same goes with values range allowed for the configuration, which can vary from a tester to another or from a model to another.

2 - The "Test points" group permits you to choose on which points tests will be done :

 : Allows you to load an equivalence table.

 : Allows you to select one or several tester points to commute to hot while processing the chosen measurement or test.

 : Allows you to select one or several tester points to commute to cold while processing the chosen measurement or test.

IMPORTANT :

The two Hot Point / Cold Point drop-down lists only references points that are truly usable for the selected measurement or test (2 wires tests, 4 wires, HT points, THT, …).

Note :

Some tests only handles Cold or Hot points, in that case drop-down lists of unavailable points (Cold or Hot) will not be displayed.

Moreover, some tests authorize choosing several points to make measurements.

3 - The "Execution" group allows to start the chosen test or measurement :
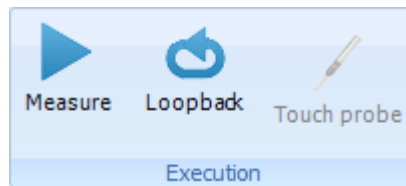
- ▶/ ■ "Measure"/"Stop" : Permits choosing and configuring measures types; but also determining if tests are validated (PASS/FAILED or valid value).

- ↻ / ■ "Loopback"/"Stop" : Permits visualizing the current measurement or the test's most important parameters (A test can be composed of several types of measures).

- ✎ "Touch probe" : Permits accessing a configuration window for the current tester; these options parametrize the context in which the tester will process every measure.

IMPORTANT :

Tests options in manual mode are shared between every tests and measures.

The test or measurement type determines if the tester handles or not some or every options.

Note :

The tests & measurements list depends on the current tester's options and on external devices usable by it; same goes concerning values ranges authorized for parametrizing which can vary from a tester to another.

2.4 - Tester

The Tester tab will allow you to handle your tester connections or carry out specific operations with it.

IMPORTANT :

Winpass 5000 is designed to work with a tester; as a result, a large part of Winpass functionality or modules can only be accessed if you are connected to a tester. If that is not the case, Winpass 5000 will only allow you to conceive projects or develop tests programs for the most part.

CONVENTION :

It is important to differentiate two connection modes to a tester :

- A **Physical** connection to a tester means that a real link (Ethernet) exists between winpass and the tester under current; in this mode, you can realize tests and measurements on products.

- A **Virtual** connection to a tester means that Winpass simulate a working tester using an image file on the disk; this mode is mainly dedicated to developers that are bound to check the good behavior of their tests programs beforehand. Winpass will generate random tests results when it proves necessary.

Note :

Connection to a physical tester will automatically save its properties in a file on the hard drive in order to check later the integrity of previously saved testers, but also to restore internal memories of a damaged tester.

In this module, functions are sorted between three groups :

1 - The "Synor 5000" group permits you to manage tester connections :
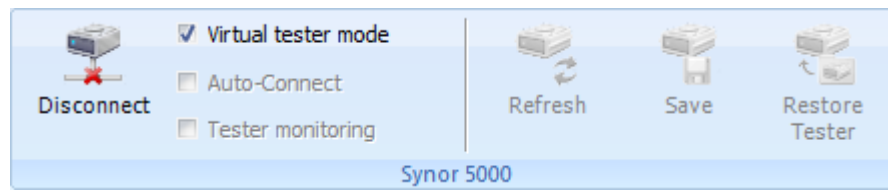
- : Allows you to try to connect to a tester or disconnect from the current one.

- : Allows you to activate virtual tester mode; this mode is only available if a tester configuration has been selected to be used as virtual tester.

- : Allows you to wait for a physical tester under current to be present when trying to connect; this mode is only active for 60 seconds, beyond that period Winpass will consider the connection to be unsuccessful.

- : Allows you to activate monitoring of the ethernet link dedicated to testers and indicate to Winpass if a tester is under current or if the connection to it has been lost; this mode permits among other things to connect to a tester only if it is truly present and available.

- : Allows you to ask a tester to refresh its current configuration (Properties and architecture).

- : Allows you to save the configuration (Properties and architecture) of a physical tester in a file in order to use it later in virtual mode.

- : Allows you to restore every information of every intelligent card of a tester whenever one or several of its memory zones has been damaged or reinitialized.

Note :

Connection to a tester (Physical or virtual) will make a "SYNOR 5000 Configuration" tab appear inside the working zone of Winpass; it will allow you to visualize properties and architecture of the current tester. In the case a physical connection, it will also allow you to modify some of its properties or choose it as the reference virtual tester.

---

2 - The "Toolbox" group permits you to carry out specific operations with a tester :
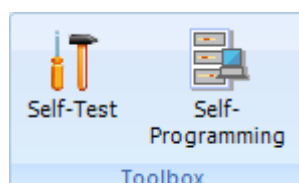
- ![icon] : Allows you to check the physical tester's intelligent cards but also every or some of its commutation cards (Test Points).

- ![icon] : Allows you to generate tests programs depending on the wiring of a product linked to a tester.

IMPORTANT :

Even if these tools are available while connected in virtual mode, they are only usable for a demonstration purpose and should not be considered as a validation process.

3 - The "Maintenance" group permits you to carry out control and integrity checks on testers :
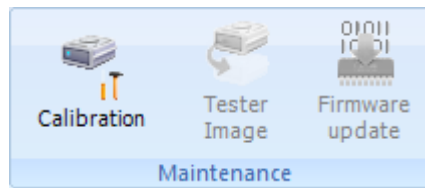
- : Allows you to verify or adjust every measuring element from a tester or display previously done calibration reports.

- : Allows you to save on disk a complete image of the intelligent cards' memory of a tester in order to restore it later.

- : Allows you to check firmware versions of the different intelligent cards of a tester or to download their latest versions.

Note :

The "Maintenance" group is only usable when connected to a physical and operational tester.

While connected to a virtual tester, the calibration function is only accessible in order to consult previous calibrations done for a tester.

IMPORTANT :

As accessing such functions can be quite sensitive, it is strongly recommended to restrain its access to a few of your maintenance employees in charge of your tester fleet (Using accounts session).

2.5 - Administration

The administration tab will permit you to set up Winpass as well as the access to its functions.

IMPORTANT :

As accessing such functions can be quite sensitive, it is strongly recommended to restrain its access to a few of your software maintenance employees (Using accounts session).

CONVENTION :

Winpass 5000 handles several software configuration files and can be set up using work sessions.

You can differentiate three configuration types :

- The **software configuration** among every Winpass users.

- The **access rights management** of users or accounts session.

- The **context of use backup** of functions common to every users.

Note :

The general software configuration is directly accessible using the "Software configuration" tab in the work zone when activating the "Administration" ribbon.

Please refer to the software configuration chapter to find a complete description of every configurable Winpass property.

In this module, functions are sorted between two groups :

1 - The "User accounts" group permits handling accounts and Winpass sessions :
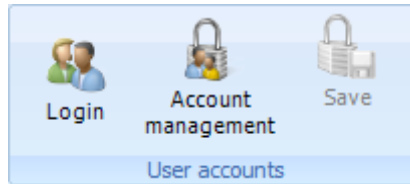
- ● ![icon] : Allows you to change account and open a new user session with different access rights.

- ● ![icon] : Allows you to handle sessions accounts; this function displays an additional tab in the work zone that will enable you to reference sessions accounts or modify its users access rights.

- ● ![icon] : Allows you to save any changes made to sessions accounts; as saving accounts changes is not automated.

Note :

Winpass 5000 being in constant development, some access rights could be added in future versions of Winpass. Please update any users accounts you created when installing a new version of this software; simply click on "Save" when starting Winpass' new version for the first time.

---

2 - The "Maintenance" group permits handling Winpass tools or optional modules licenses :
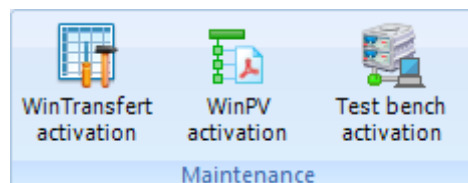
Up to this day, three complementary modules has been included in Winpass : WinTransfert, WinPV and Test Bench management.

Tools and modules managed by this license system are commercial options. You need to make a specific request to our sales department for every option in order to use it.

Note :

A license consists of a serial number linked to an activation key. A license can be used on as many workstations as needed; however, it is recommended to only activate such modules on workstations that will truly use the module.

2.6 - Help

The help tab permits you to gain access to different manuals, handbook and guide from Winpass without using paper format or PDF.

In this module, functions are sorted between two groups :

1 - The "About" group allows you to visualize release notes of the installed Winpass version.
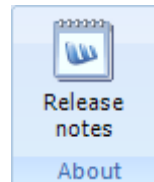
2 - The "References" group allows you to choose and load a guide or manual with the possibility to make key-words researchs.
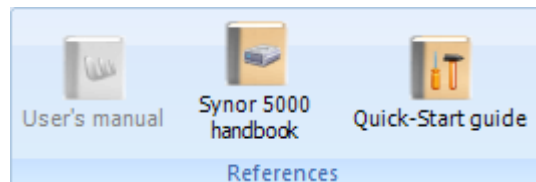
The current guide is directly accessible using the "Help" tab displayed in Winpass work zone.
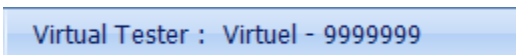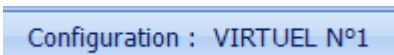
NOTE :

The current guide or manual is accessible to every Winpass user whichever the current module is.

---
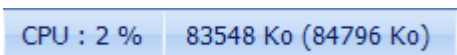
3 - Status Bar

The Winpass status bar located at the bottom of the screen permits quickly visualizing information about the current tester and the computer usage. Four panels relate to testers, four relate to the computer.

Hence, from left to right, you can visualize the following testers information :

- : These first two panels relate to the link between Winpass and a tester; they indicate in particular if a connection is successful, partial, failed or lost.

- : The third panel shows which tester is connected to Winpass (Name & serial number).

- : The fourth one provides the tester's configuration name (Architecture) detected by Winpass if it has been saved on your hard drive.

Then, still from left to right, you can see the following information about your computer :

- : The first two panels show CPU resources and memory used by Winpass.

- 06/09/2016   11:57:04 : The two last panel state the current date and time.

4 - Projects Panel

This panel, always located on the left side of the screen, shows you currently edited or executed projects, depending on the active module (Edit or Execute).

Projects are displayed in a tree structure of dependencies between files; thus, an equivalence table is linked to a tests program, a tests programs is linked to a tester configuration and a tester configuration is linked to a tester.
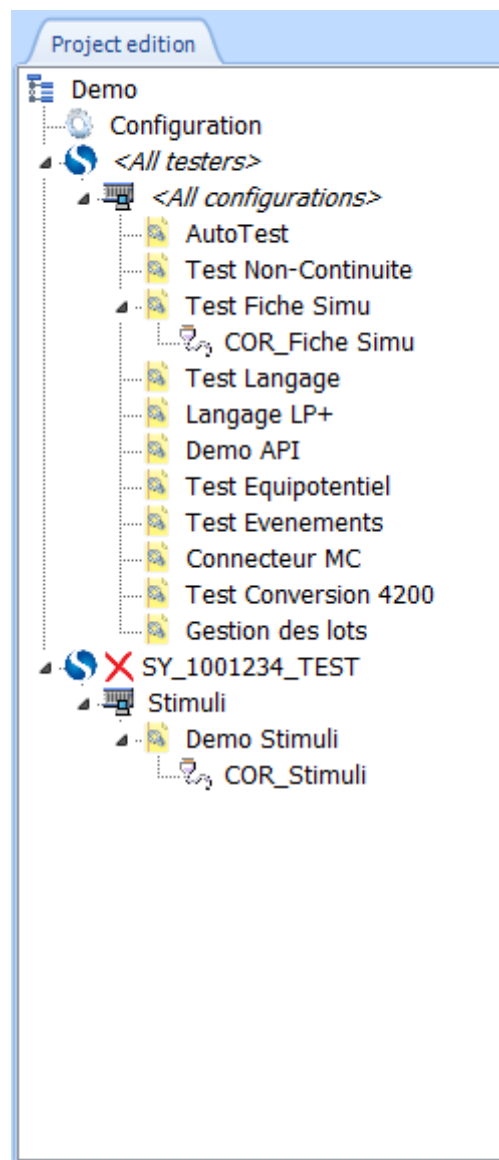
In execution, this structure will always be frozen and will always display truly usable files.

In edition, you can modify this structure by renaming, moving, adding or deleting files.

A **single click on an editable file** (In Edit or Execute mode) from the project will automatically display it in the work zone if it is open.

A **double click on a file** will load it in the work zone if it is editable or open the associated tool if possible (For instance, the project execution configuration).

> **NOTE :**
>
> At any time, indications "[Edition]" and "[Execution]" in the project panel will let you know whether the displayed project is opened in the edition or execution module.

5 - Tool Panel

Always located on the right side of the screen, this panel lets you access tools related to editable files. In edition, there is as many tools as editable files types, while there is only one tool in execution.

5.1 - Edition Tools

Edition tools provides help with creating or modifying editable files such as tests programs, equivalence tables or dialog boxes.

Please see chapters related to tests files edition for more details about creation help tools.

5.2 - Execution Tools

The execution tool gives tracking information about the currently executed tests program, as well as states of the different tester options (security loop, lights, remote, API, …).

Please see chapters related to tests execution for more details about execution tracking.

6 - Messages Panel

Always located at the bottom of the screen, this panel lets you see the different messages generated by Winpass while verifying/compiling editable files (In edition or execution) as well as while executing tests program.
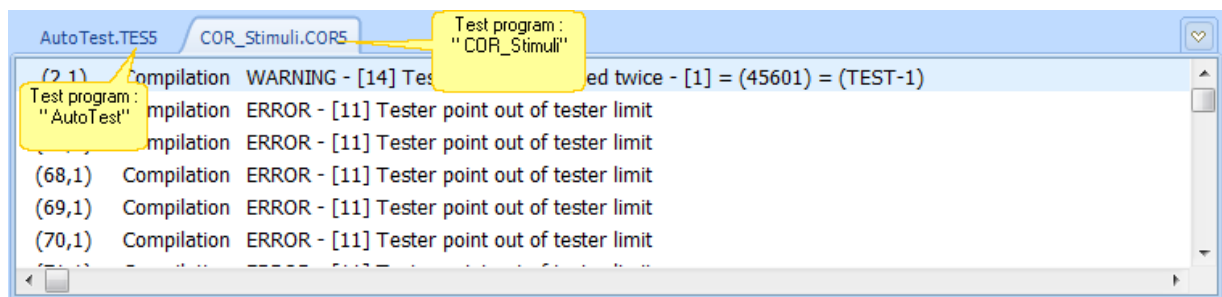
The message panel references as many tabs as there are open files in edition or execution; however, only execution or edition messages are displayed depending on the currently active module (Edit or Execute).
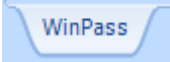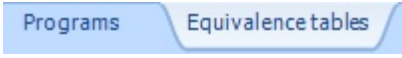
A **single click on a message tab** will display the associated file in the work zone.

A **single click on a message line** will align you directly on the corresponding file line whenever possible; In execution, this will align you on the associated test result line.

7 - Work zone

Located in the middle of the window, Winpass work zone is certainly the most import zone as it references, among others, every files that are currently being used.

A first batch of tabs located at the bottom of the zone regroups files by families :

- **WinPass** : The "Winpass" tab gathers every configuration documents or files dedicated to the software operation.

- **Programs** **Equivalence tables** : The "Programs", "Equivalence table", "Dialog box" tabs pool files currently being edited depending on their types.

- **Execution** **Test Results** : The "Execution" and "Tests Results" tabs bundle, on one hand tests programs currently executed, and on the other hand execution reports linked to every tests programs.

7.1 - Winpass Tab

The "Winpass" tab of the work zone is the main tab of the software, letting you access data and properties managed by Winpass 5000.

Hence, it is in this tab that you can visualize the software configuration, users accounts definition, but also the current tester's configuration; this information is displayed only when the given function or module is active.

Two other tabs, "Welcome" and "Help" are always accessible, anytime.

7.1.1 - Welcome Tab

The "Welcome" tab is the first visible when starting Winpass; It will let you visualize historic of each operations previously done using Winpass.

It enables you to load a new file from the historic (in edition or execution), but also to define some files as "favorites" when you wish to work with it regularly.

7.1.2 - Help Tab

The "Help" tab will lets you read the current manual but also to make some researches in it.

7.2 - Editing Tabs

The editing tabs, three thus far, permit classifying documents currently edited depending on their types (Programs, equivalence tables, dialog boxes).
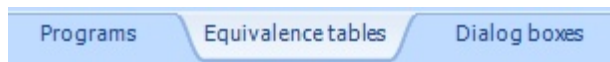
IMPORTANT :

These tabs are only accessible whenever Winpass is in edition mode; which means whenever the execution mode is not active (Also see Execute ribbon).

7.3 - Executing Tabs

The execution tabs, which are currently two, permit on one hand to visualize tests programs loaded in execution and on the other hand to see the last tests results report of every executed program.

IMPORTANT :

These tabs are only accessible whenever Winpass is in execution mode; which means whenever the editing mode is not active (Also see Edit ribbon).

8 - Interactions and links between panels from the interface

As Winpass 5000's interface is rich information wise, the software automate some navigation operations in order to make your everyday work easier.

To that end, **a strong link exists between the different interface panels** (Project, tools, messages, edition zones, execution zones).

> **CONVENTION :**
> Interactions with an interface panel will affect other panels so that Winpass always displays the right information at the right time.

For instance, choose a tests program in the projects panel will automatically switch to the corresponding message panel and the edition or execution tab linked (And in chain reaction, the corresponding edition or execution tool will be displayed and updated).

On the contrary, selecting the message tool of a specific document will display the corresponding edition or execution tab (with updated linked tool) and will choose the document inside the project if possible.

Furthermore, going from a tab from a ribbon to another can also affect the view and update several interface panels.

> **CONVENTION :**
> Winpass 5000 has three work modes : tester mode, edition mode and execution mode. These three modes correspond to the "Tester", "Edit" and "Execute" ribbons.

Hence, **activating any of the "tester", "edit" or "execute" ribbon will lead to a switch of Winpass work mode**, which has for consequence to display the currently edited project, or the currently executed project, or to hide the project panel in the case of the tester mode.

Switching mode using the tabs from the ribbon can also imply, in chain reaction, an update of other interface panels when necessary.

Chapter
Chapitre
Kapitel

IV

Basic Functions

IV - Basic Functions

This chapter aims to familiarize you with Winpass 5000' modus operandi. It describes main functions that you will need to use one day or another in order to achieve your products' tests.

It addresses notions of security and access rights, testers management, test documents redaction, integrity and tests validation and finally the implementation of test in real conditions.

For more details about possibilities given by Winpass 5000 and the usage of modules or more specific tools, you should see the following chapters that will describe more precisely and technically some notions overflown in this chapter.

1 - Opening a Session

Since its development, Winpass 5000 focuses on modularity of configuration and usage of its tools and functions that it offers to stakeholders that wants to conceive or realize products tests.

For that, **accessing Winpass is achieved using a session system**; which aims at enabling to a specific user the only functions and tools that he will need to use.

1.1 - Starting Session

Opening a session is mainly done when starting Winpass and uses a user rights account manager.

NOTE :

Without going through details, a user account is defined by three properties :

- An **account name** to differentiate accounts between them.

- A **user name** that permits differentiating operators bound to make some tests.

- A **password**, optional, when you need to secure access to certain Winpass tools.

Furthermore, a **user account describes every access right** to function and/or operating modes of tools that will be managed while opening a session.

When opening Winpass a dialog box will pop up asking to enter information about an existing account in order to create a session and configure Winpass' interface depending on its associated rights :

IMPORTANT :
The opening session window allows for three tries. After that, Winpass will shutdown and you will have to start it again.

Reminder, **you are not obliged to define a password for your accounts**; It is on you to secure access to specific functions that you judge sensitive when operators using Winpass are not trained to use potentially hazardous tools.

NOTE :

Winpass 5000 is installed with a demonstration account by default so you can have a first access to the software, familiarize yourself or train operators bound to use it.

To start a "Demonstration" session, all you need to do is entering "demo" as Username and click on OK. This account does not require a password.

This account is designed to be erased and replaced by more restrictive ones in terms of rights when using Winpass for testing real products under production.

1.2 - Switching Session

Once a session is open and if its access rights allow it you can switch session without closing Winpass using the "Administration"/"Login" ribbon.
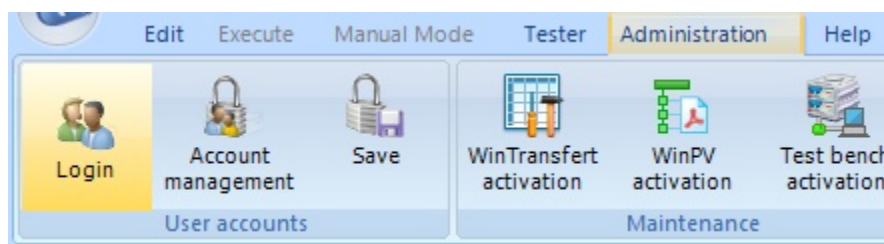
NOTE :

If switching session fails (after three tries) or is aborted, the previous session will still be active and Winpass will not be closed (Contrarily to failing to open a session when starting Winpass).

---

2 - Connecting to a Tester

Winpass 5000 aims at achieving your products' test while using our testers (Synor 5000 line), and when appropriate using measurement tools and/or external devices referenced by the tester.

Even if Winpass lets you conceive projects and tests programs without being connected to a tester, being linked to a tester (and more particularly using a virtual mode) will bring you an easier way of conceiving and editing your tests documents.

Indeed, **different edition tools handle most tester characteristics** in terms of architecture (Number of usable tests points), measurement cards (available tests instructions) but also external devices linked to the testers (extra tests types).

Connection to a "Physical" or "Virtual" tester is done using the tab of the "Tester" ribbon :
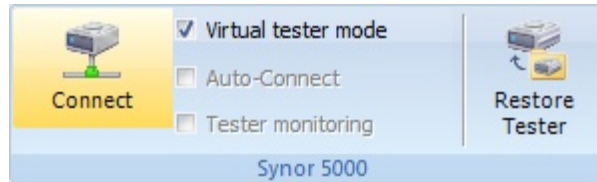
NOTE :

Whenever a connection is established with a tester (either physical or virtual), Winpass displays the "SYNOR 5000 Configuration" tab inside the work zone (In the "Winpass" tab). This tab will give you every information about your tester and you will gain access to new tools in "Tester" tab of the ribbon.

2.1 - Physical Connection

Connecting to a "physical" tester is the prior requirement when using Winpass for the first time, as throughout this connection phase Winpass will compare characteristics (Name, serial number, architecture, …) from the currently linked tester (via Ethernet) with configuration files saved on your hard drive.

IMPORTANT :

If no tester configuration matches with the physical one, you will be asked (using a dialog box) to give this configuration a name if you consider it valid and Winpass will then save it automatically on your hard drive.

[IMAGE boite de dialogue nom de la configuration testeur]

This method shows three advantages :

- It allows for **reporting any unexpected change** of one or several tester characteristics and either validate or not its integrity before testing your products.

- It **permits Winpass to re-use tester configuration files** in order to emulate testers' properties **when using virtual mode.**

- Where appropriate, it **references the properties of external devices** linked to a tester and check its compatibility with it.

During this connection phase, in addition to establishing the link with the tester, Winpass will try when appropriate to connect to the different external devices referenced in the tester.

The Winpass status bar (located at the bottom of the screen) will give you information about the process of connection establishment; the following table describes its main phases and states :

[TABLEAU barre d'état de progression]

2.2 - Virtual Connection

Virtual connection to a tester uses tester configuration files present on the disk and stored in the testers directory pointed by Winpass software configuration.

CONVENTION :
The default storage directory for tester configuration files is as follow :
[Winpass installation directory]**Testers**
File names are always the same shape as follow :
SY_*[Tester serial number]_[Tester name]*.**CFG**

A tester used in virtual mode is a fair view of a physical tester in terms of characteristics (Architecture, options, available cards and functions, external devices referenced) and can be used to perform tests programs. However, Winpass can only mimic its operation :

- Tests results are randomly and partially generated.

- Some Winpass tools are only usable for demonstration purposes (Self-Test, Self-Programming, …).

- Some tester functions are not accessible (Touch probe, firmware update, …).

Where required, external devices referenced by a virtual tester will work in virtual mode too and their operation will be simulated; However their own characteristics will be kept as accurately as possible.

IMPORTANT :
Winpass only handles one tester configuration at the time; Which can be chosen in the "SYNOR 5000 Configuration" tab of the work zone. You have to be connected to a tester (either in virtual or physical) to choose a different tester configuration.
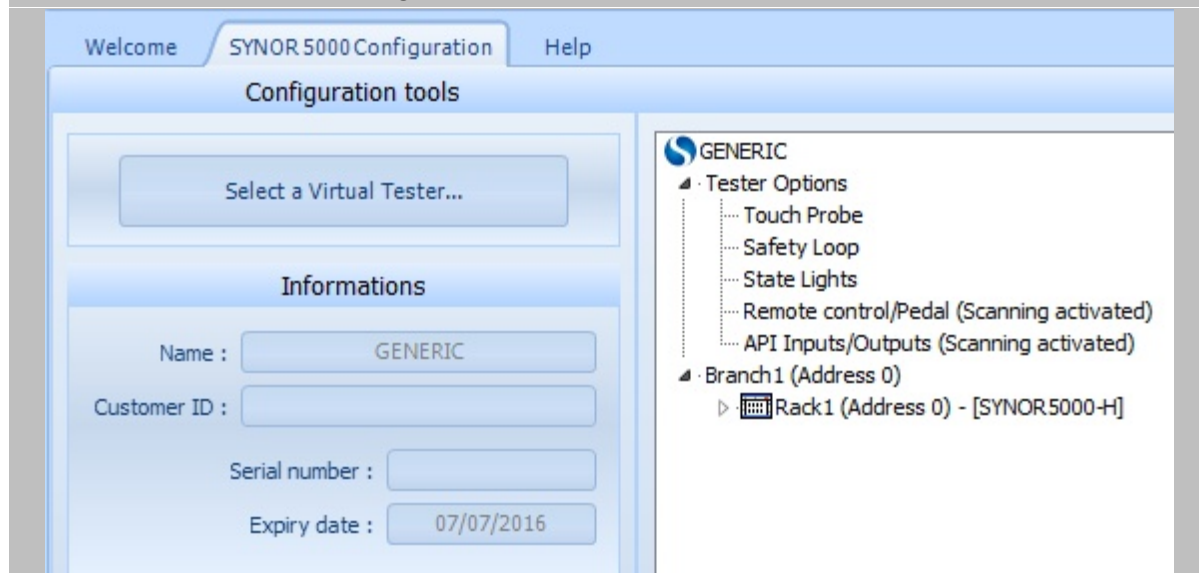
NOTE :
You can use Winpass 5000 to develop or correct your projects and tests programs on any computer using any tester configuration.

For that purpose, you only need to copy configurations storage directory (…\Testers\) on every computer you want to develop on. However, this method only permits having tester configuration images at a given time and offer no validity guaranty over time.

The safest way consists of stocking tester configuration on a network disk and configure every workstation with Winpass installed to set their configuration storage directory on the same network path.

In order to connect in virtual mode, you only need to check the "Virtual tester mode" box before clicking on the "Connect" button :
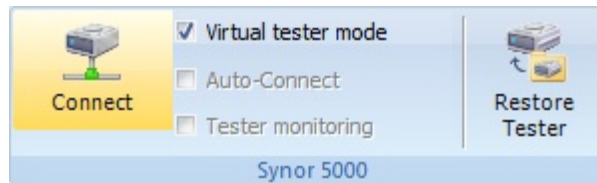
A virtual connection only fails when no virtual tester configuration is defined or if the corresponding file cannot be loaded from the storage directory (Also see "Software Configuration").

An established connection in virtual mode is reported with a yellow light in the status bar :
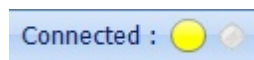
3 - Editing Test Files

In order to test your products you will need as a first step to write every file mandatory for its validation.

The essential file handled by Winpass 5000 is the tests program. It will allow you to describe every test and measure you think is needed to either validate or not your product.

The Synor 5000 line mainly manages matrix (commutation cards with relays) composed of tests points identified by a numerical rank. These numerical values having nothing to do with your products directly, Winpass 5000 allows you to link a label to each of the tester's point in your product in a file named an **equivalence table** (or corresponding table)**.**

Finally, your projects or products being potentially composed of sub-groups or sub-products, Winpass 5000 implements a **tests projects manager** that will allow you to organize and break down your tests when your products need a step by step validation, or when you want to gather a whole line of products in the same project.

CONVENTION :
A **tests program** regroups every tests/measures and process needed for a product or sub-product validation.
An **equivalence table** references every matrix point from a tester used to test a product with an alphanumeric label.
A **transcoding** is the usual representation of a numerical test point. It is described in an equivalence table and used in a tests program.

3.1 - Creating a Test Project

Winpass 5000 can only complete your products tests using tests projects. These can link several files and keep this relation over time as a project file.

Its prime objective is to **save the strong connection that exists between an equivalence table and one or several tests programs** using labels for tests points instead of their numerical ranks in a tester.

A tests project manage other types of relations and other types of files that will not be discussed in this chapter. For a deeper knowledge of tests projects management, please see the advanced use chapter concerning projects.

> **CONVENTION :**
>
> A **tests project** regroups every references used to complete a product line's validation or break down products validation between sub-parts or sub-products.
>
> A **project reference** is the document path saved on your hard drive. A path is only an image pointing on a file and brings no guarantee of the file or the storage directory's existence.
>
> A **project element** shows the relation that exists between one or several references of the same project.

3.1.1 - Creating a new project

Creating tests projects is done using the "Edit" tab of the ribbon; the "Project" group of this tab contains every tools needed to conceive and manage projects :
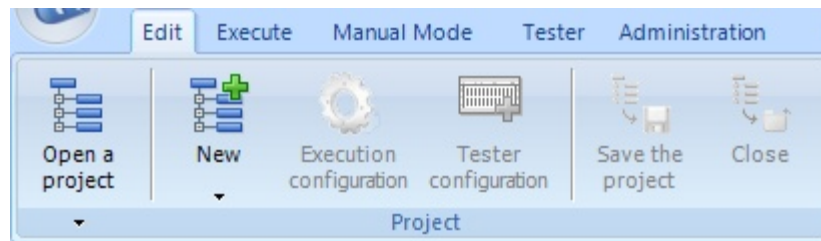
In this group, the "New" button allows you to create a new project but also to create new files that will be associated to the current project. This button, composed of two parts, permits constructing a project in two different ways :

    - Clicking the upper part of the button will give you access to a helping tool for creating projects

    - Clicking the lower part of the button will allow you to manually construct a project and to add new documents to it.

3.1.1.1 - Adding a new project

When choosing "New test project", a dialog box will ask you to enter your project's name, the same goes when adding new programs or corresponding tables to your projects.
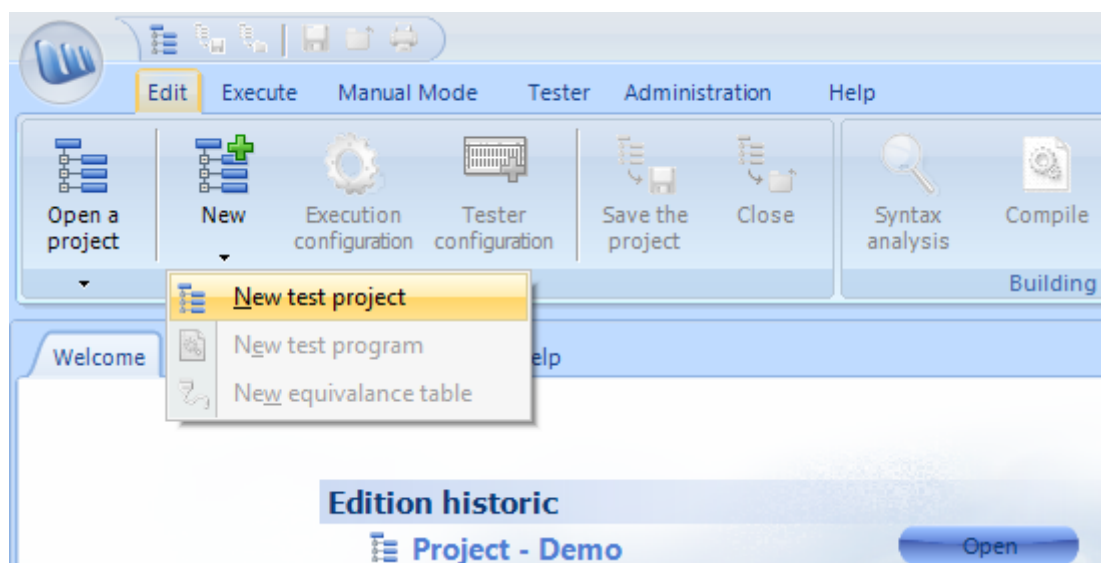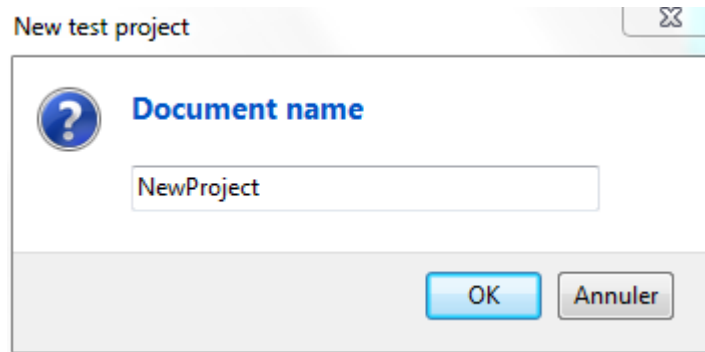
The new project is then displayed in the projects panel (Left side of the screen).

> **IMPORTANT :**
>
> **A test project name has to be unique** as every file generated by Winpass 5000 are stored in a common directory.
>
> When creating a project Winpass first checks if no project bearing the same name exist in the storage directory; otherwise Winpass will not create a new project but load the existing test project in the project panel.

3.1.1.2 - Adding a new tests program

You can then add a new test program to you project using the same menu (clicking the lower part of the "New" button) and choose its name.
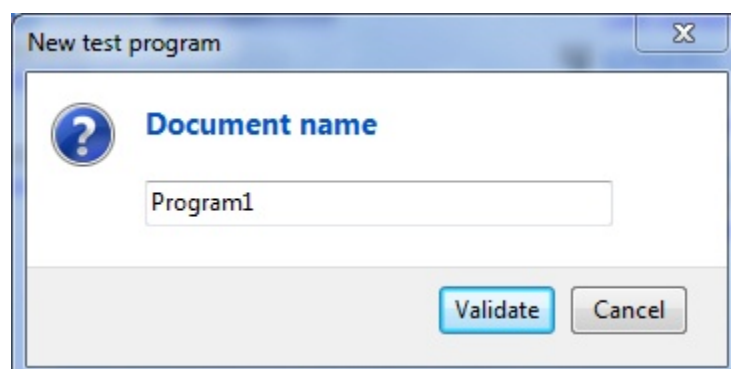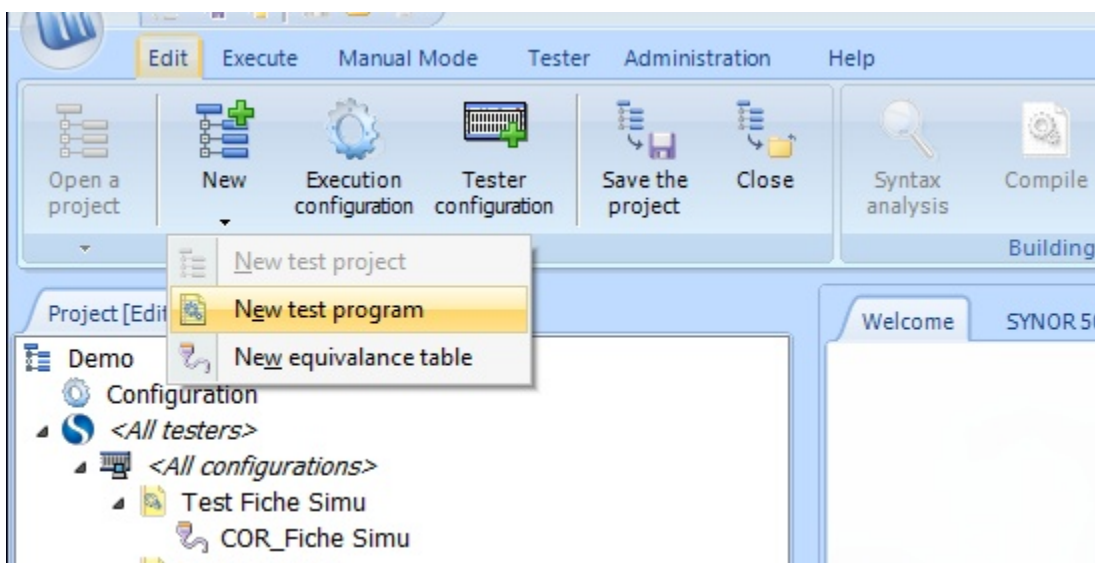
This last action has several effects on Winpass :

- A *Tester* element, by default "*<All testers>*" , is added in the project tree structure (Project panel).

- A *Configuration* element, by default *"<All configurations>"* is added under the *Tester* element.

- The new tests program is added into the structure, under the tester configuration by default.

- A new edition file is created in the work zone in the "Programs" tab.

- The dialog box associated to the program editor is displayed in the messages panel.

- The program creation help appears in the edition tools panel (right part of the screen).
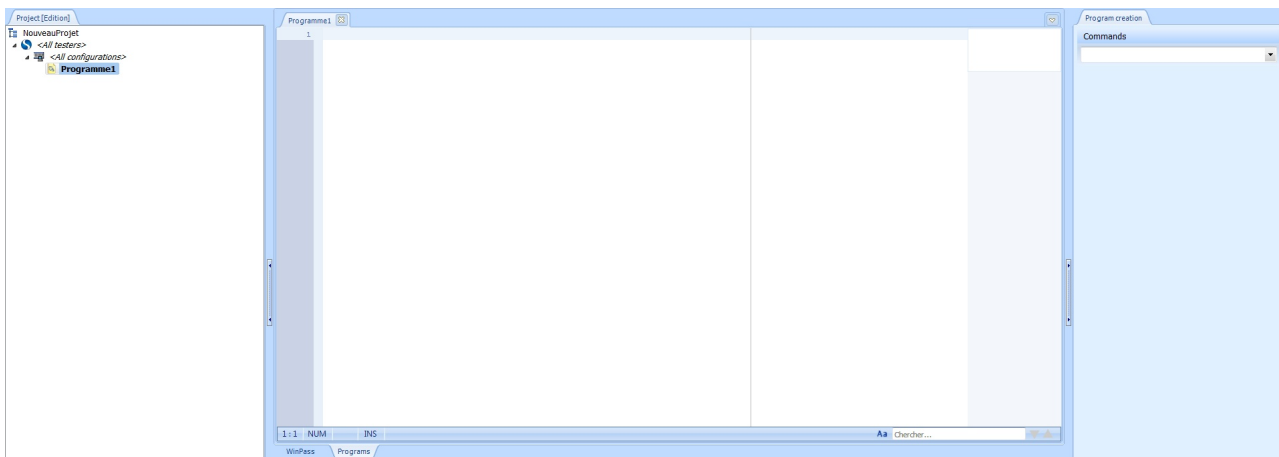
CONVENTION :

A test program is always linked to a tester configuration even if testers or configuration elements references non-identified testers or configurations (meaning a tester or configuration by default).

The chapter about Winpass' advanced execution dedicated to project management describes the way one or several test programs can be dedicated to a specific tester configuration but also reference several testers inside the same project.

IMPORTANT :

A test program name has to be unique in the same project as every program file generated by Winpass 5000 are stored in a directory common to the whole project.

When creating a program Winpass first checks if no program bearing the same name exist in the project storage directory; otherwise Winpass will not create a new program but add the existing test program in the project panel.

Nevertheless, **this does not forbid a test program to appear several times in a project.**

This also implies that **a test program can appear in several projects** (Under certain conditions).

3.1.1.3 - Adding a new equivalence table

Using the same menu (Click on the lower part of the "New" button), you can now add and join a new equivalence table to the previous test program and choose its name.
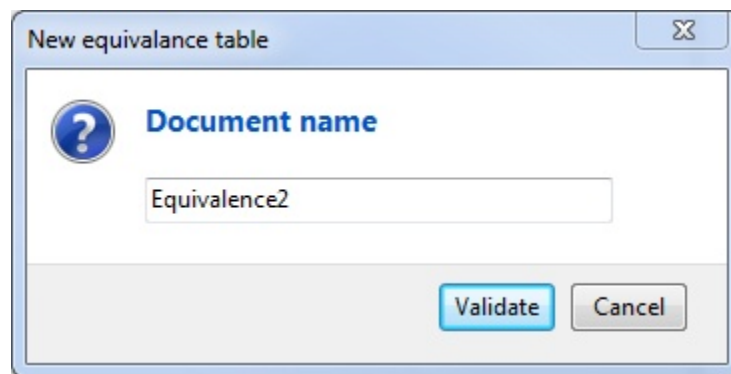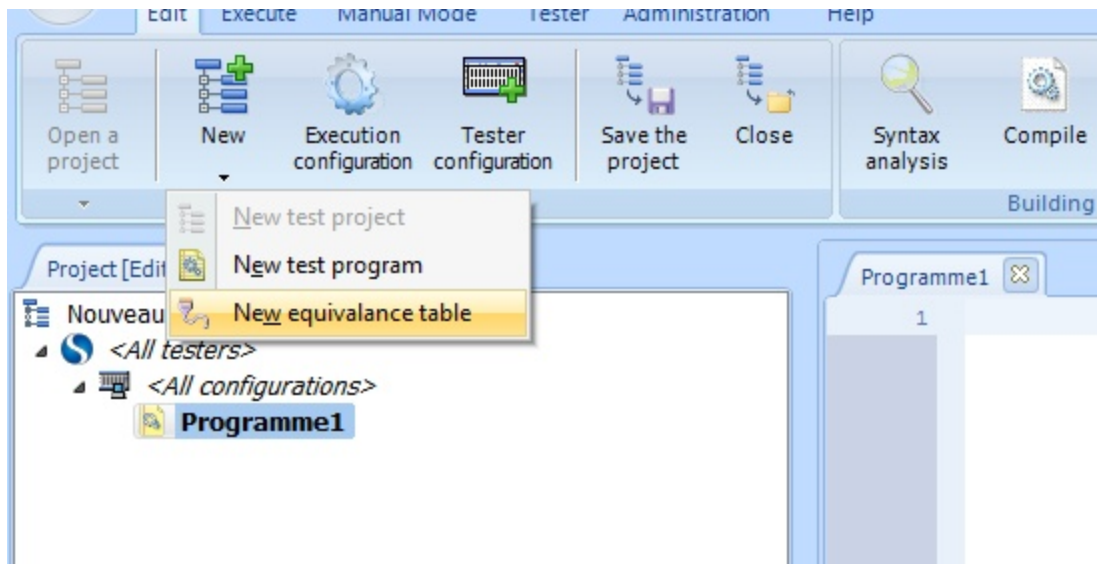
This last action has several effects on Winpass :

- The new equivalence table is added to the project structure under the current test program.

- A new edition file is created in the work zone, in the "Equivalence tables" tab.

- The dialog box associated to the equivalence table editor is displayed in the messages panel.

- The equivalence table creation help appears in the edition tools panel (right part of the screen).
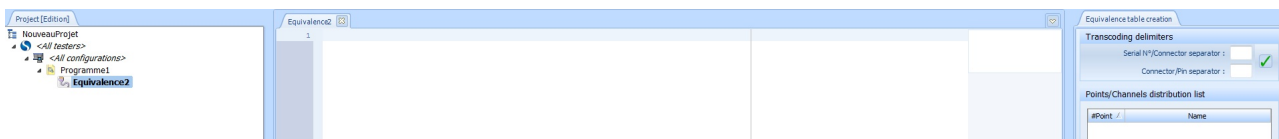
CONVENTION :

Adding an equivalence table in a project can only be done if a test program is selected in the project structure and has no other equivalence table linked to it.

Otherwise, adding an equivalence table will not be taken into account and no file will be added nor loaded.

As a reminder, equivalence table are optional in a project; They are reference files for conceiving test programs and generating test execution results reports.

> **IMPORTANT :**
>
> **The equivalence table's name has to be unique in a project** as equivalence tables files created by Winpass 5000 are stored in a common directory for the whole project.
>
> When creating a table Winpass first checks if no table bearing the same name exist in the project storage directory; otherwise Winpass will not create a new table but add the existing equivalence in the project panel.
>
> However, **this does not forbid an equivalence table to appear several times inside a project.**
>
> This also implies that **an equivalence table can appear in several projects** (Under certain conditions).

3.1.2 - Project Creation Help Tool

The help tool for projects creation will guide you while conceiving simple tests projects. It allows you to carry out several actions using a single interface by breaking down projects creation step by step.

This tool is activated by clicking in the upper part of the "New" button of the "Project" group.
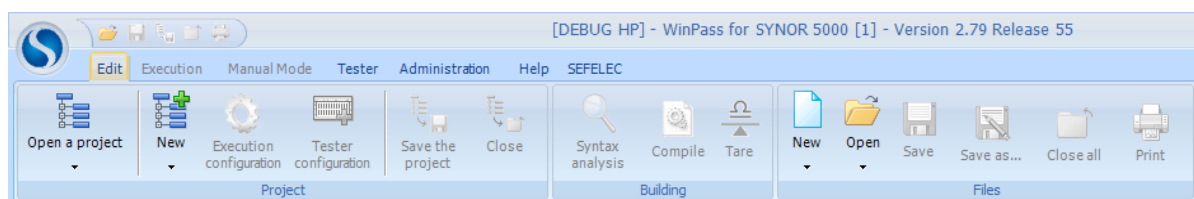
The upper part of the tool gives you information about the current project creation step, the lower part gives you the choice of actions doable while creating (Project construction, navigation from one step to another or canceling the project creation).

The project creation requires two to four steps. Starting from the second step, you can launch the project's construction without being forced to proceed to the last step.

Once the creation is validated ("Build" button), Winpass automatically creates the new project, tests programs, add existing programs and save it all in the Winpass 5000 project directory.

3.1.2.1 - Properties of the Project to Create

Winpass firstly displays a windows letting you select project properties and creation options :

The "Automatic project name" options gives the project the same name as the 1st test program in it.

You can also import test programs from older tester versions (Winpass & Synor 4200) by converting them in the Winpass 5000 version of this type of files.

Finally, you can possibly indicate to which tester is dedicated this project.

3.1.2.2 - New Test Programs to Create

The 2nd step consists of choosing potentially the names of the new tests programs to create and to which testers they have to be linked :
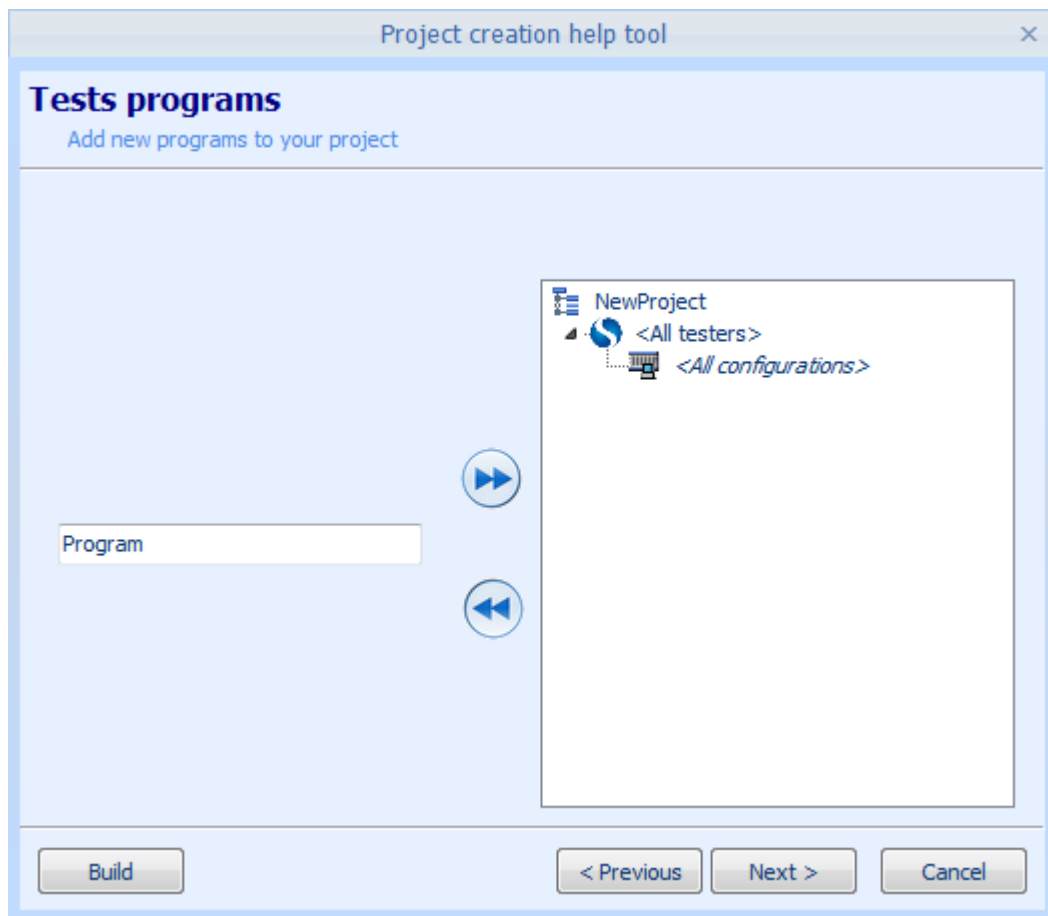
In this window you can enter the test program's name and then, using the "Drag & Drop" method, you can move this name on the tester configuration. You can repeat this operation as many times as you wish to create that many new test programs.

3.1.2.3 - Existing Test Programs to add to the Project

The 3rd step consists of possibly adding existing test programs into the new project and setting to which tester they are linked to :
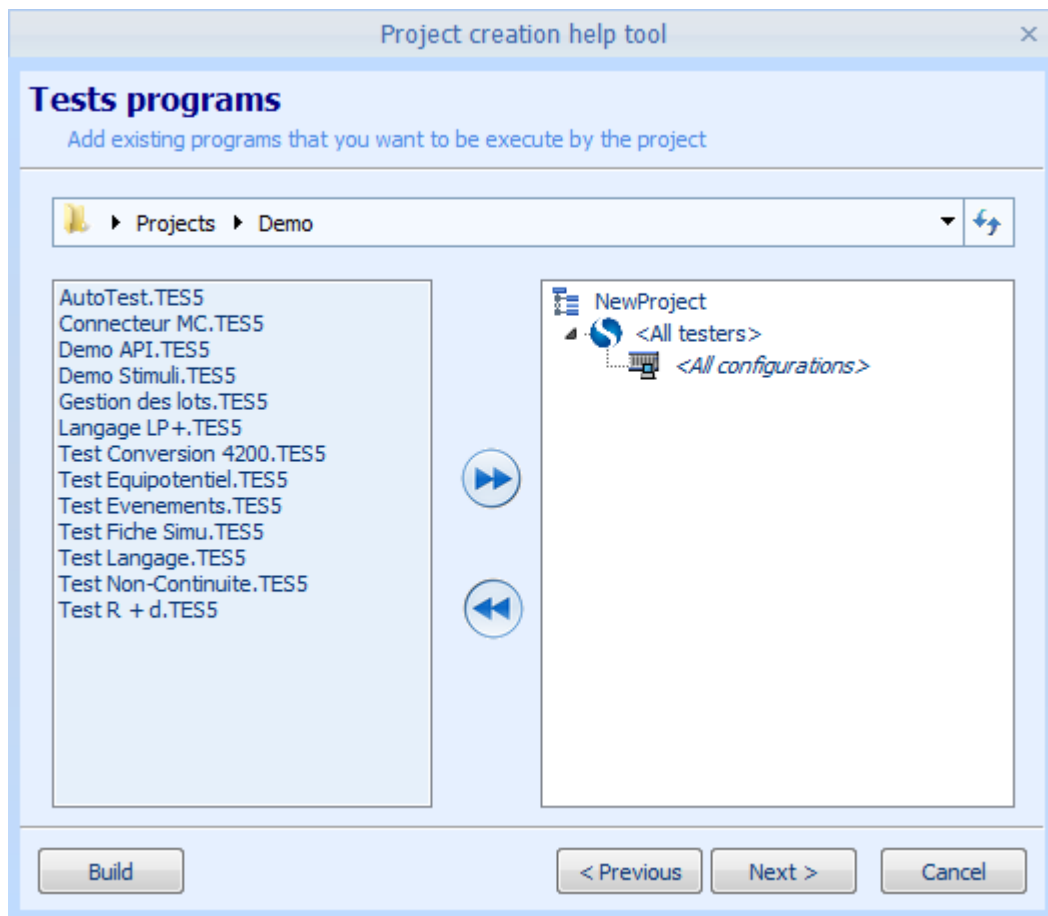
In this window you can choose in which directory to find the test programs you want to add and then, using the "Drag & Drop" method, you can move a file on the tester configuration. You can repeat this operation as many times as you wish to import that many existing test programs.

3.1.2.4 - Choice of equivalence tables linked to programs

The fourth and last step consists of possibly associating the equivalence tables mandatory for the test programs' execution (New or existing) :
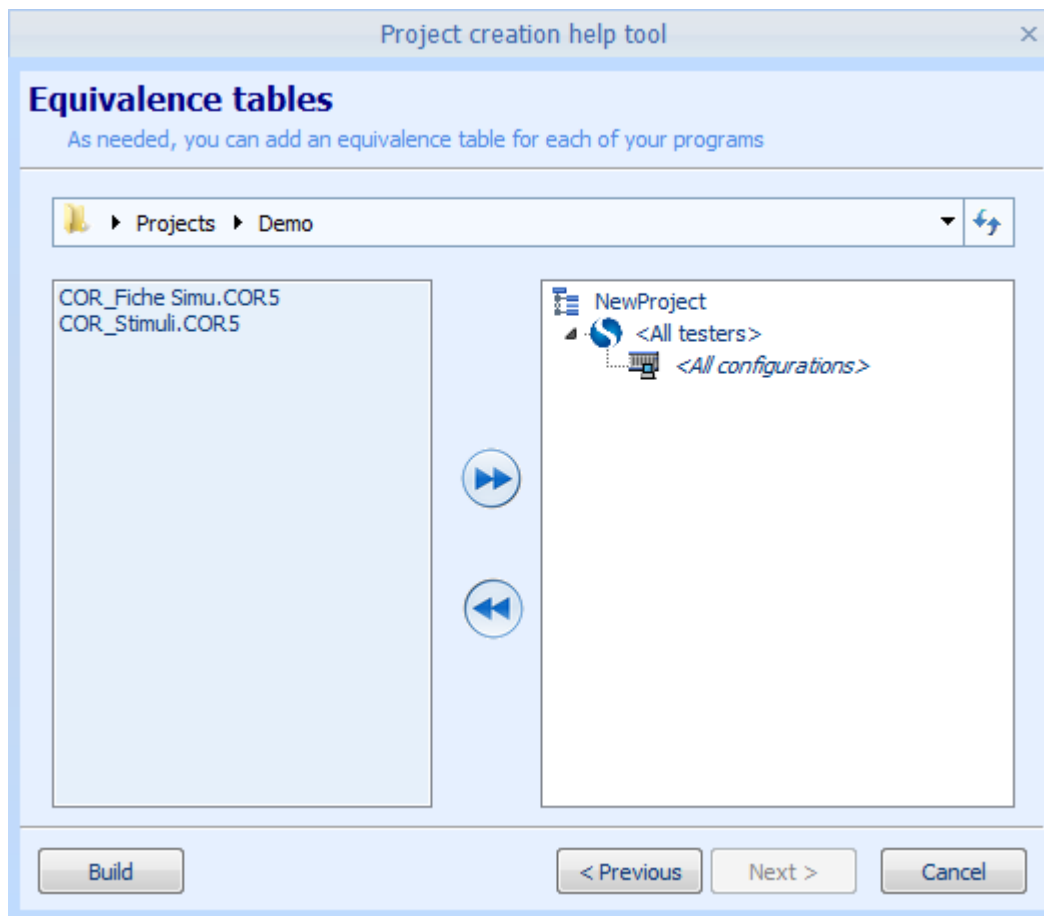
In this window you select the directory in which are located the equivalence tables you want to use then, using the "Drag & Drop" method, you move an equivalence table file on the test programs that uses it. You can repeat this operation as many times as you wish to link that many existing equivalence tables to test programs.

3.1.3 - Management Tools (Add/Rename/Remove/Move)

The projects panel allows you to visualize the links between the different elements and files inside the same tests project. It also permits you to achieve certain actions to update or complete your existing projects.

NOTE :

The only possible way to modify a project is while in edition mode.

In execution mode, the only actions available from this panel are selecting and loading a test program.

In edition, the project panel lets you access a **contextual menu by right clicking on an element** from the current project. This menu allows for four basic operations :
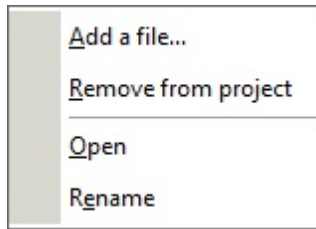
In the projects panel you can also **easily move tests files using the "Drag & Drop" method**.

IMPORTANT :

Actions modifying a project don't necessarily update the project file on disk. You need to save the project or ask for it to save while closing for every change you made to be definitive.

3.1.3.1 - Adding a file to the Project

The "Add a file..." action from the contextual menu allows you to add a file reference (Path on disk) to your project depending on the selected element's type. A dialog box will ask you to choose which file to add to your project :
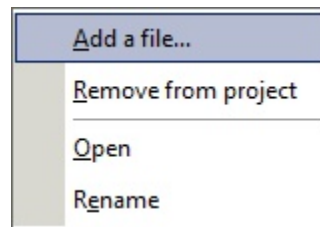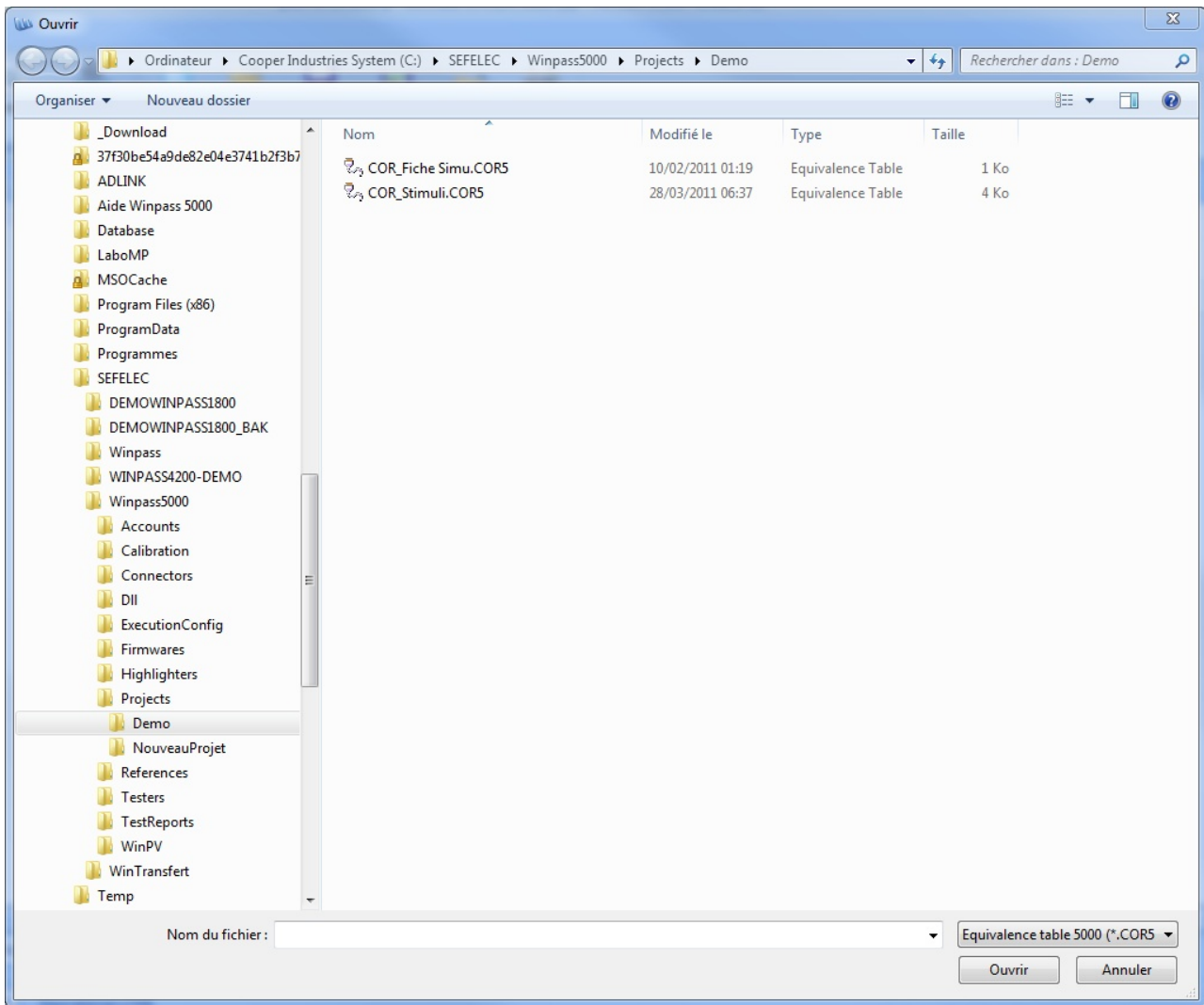
Thus, if the pointed element is either of Tester or Tester Configuration type, you will only be able to choose tests programs related to this tester.

In the same way, when the pointed element is a test program, you will only be able to choose equivalence tables.

> **IMPORTANT :**
> When adding a file whose path is not the same as the current project directory being edited, its reference (Path on disk) will be modified when saving the project.
> This file ("Source" Document) will be copied in the project directory and its reference in the project structure will point on that copy of the original file.

3.1.3.2 - Removing a file from the Project

The "Remove from project" button from the contextual menu allows you to remove the current reference (Disk path) of the pointed file from the current project. A dialog box will then ask you if you also want to delete the said file on the disk.
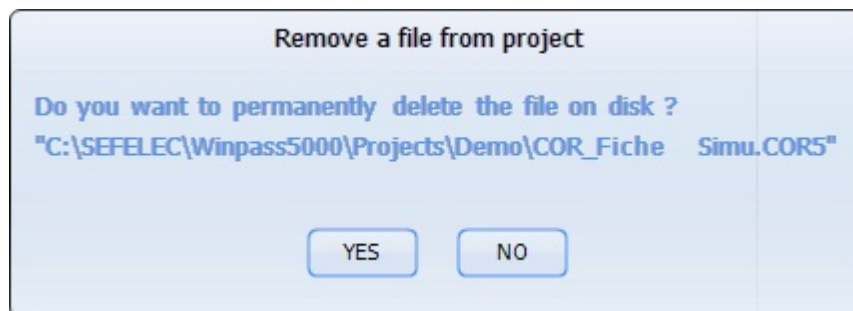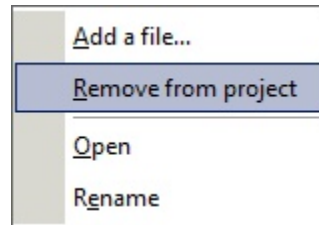
IMPORTANT :

When permanently deleting a test file on disk you need to make sure that there is no other allusion to that file in the current project.

3.1.3.3 - Renaming a File

The "Rename" button from the contextual menu allows you to change the name of the pointed file. It is directly modifiable in the project structure. Once the new name is validated, Winpass also renames the file on the disk.
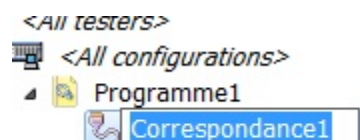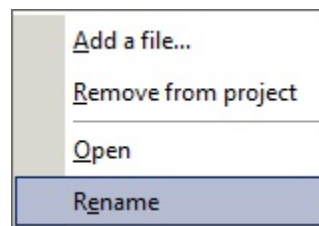
IMPORTANT :

When renaming a test document you need to make sure there is no other allusion to this document in the current project.

When the new name already identifies a file on disk (Even if there is no allusion to this new name in the project), the new name is rejected and the old one kept.

3.1.3.4 - Moving and Scheduling documents

Moving elements in a project can prove useful if you want, for instance, to switch tester in order to execute a test program, or to schedule differently the project's test programs execution.

Moving elements inside a project in edition is done by Drag & Drop :

1- Hold down the left click on the pointed element.

2- Move your mouse while holding down the left click.

3- Release your left click wherever you want to place the element.

While moving the element, the mouse cursor changes shape in order to signal whether placing the element here is allowed or not.

> **NOTE :**
>
> When moving an element inside the project structure, you move de facto every one of its underlying element (For instance, moving a program linked to an equivalence table).
>
> You can also schedule Tester elements inside a project, and the Tester Configuration elements order inside the same Tester element.

3.1.4 - Saving Projects

Test projects are stored on your hard drive inside the project directory pointed by Winpass software configuration.

In this directory, common to every test projects, Winpass stores every file linked to the project in a sub-directory bearing the same name as the project itself.

> **CONVENTION :**
> The default storage directory for test projects is :
> $\qquad$ *[Winpass installation directory]***Projects**[*Project name*]\\

> **NOTE :**
>
> Saving a project presents the advantage of automating the update of every tests files linked to it.

3.1.4.1 - Manually Saving Projects

Manually saving test projects can be done anytime during their conception; To do so, all you need to do is click on the "Save the project" button in the project group :
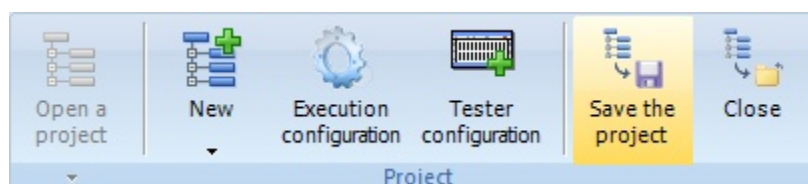
When saving a project, every new or modified test files (Programs and equivalence tables) will be saved as well in the project directory.

Test programs and equivalence tables added to the project from another project (Meaning loaded from another project's directory) are copied in the current project's directory when saving.

> **IMPORTANT :**
>
> Copying a test document from a project to another implies a total interruption of the link that could exist between two test files bearing the same name and appearing in two different projects.
>
> This means that even **if two test files bear the same name in two different projects, modifying one of them will not modify the other one (although identical at the beginning) bearing the same name and present in another project.**
>
> This management choice has been motivated with safety and validity of tests result reports generated while executing tests programs.
>
> Indeed, if a unique program was referenced in several tests projects, every modification made to it (Removing a test, modifying a measure parameter, ...) could potentially cause some flawed test results for every other project using the same program without you ever noticing it.

3.1.4.2 - Closing a project (Semi-automated update)

Closing a test project in edition is done by clicking the "Close" button in the "Project" group :
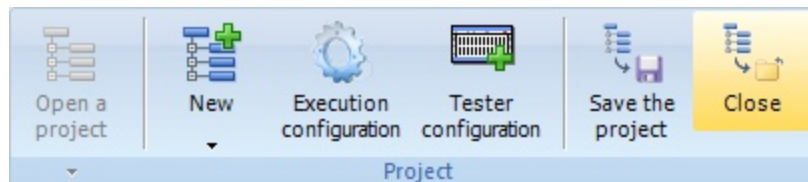
Closing a project leads to several verifications :

- First, Winpass checks if modifications have been made to the project; In that case, a dialog box will ask you whether you want to save these changes or not.

- Then, Winpass checks if test files in the project are found in the project directory; If not, files not found will be automatically saved.

- Finally, Winpass checks if modifications have been done to test files under edition in the project; In that case, a dialog box will ask you whether you want to save these modifications of if you want to discard them.

Dialog boxes that can be displayed while closing are of two types :
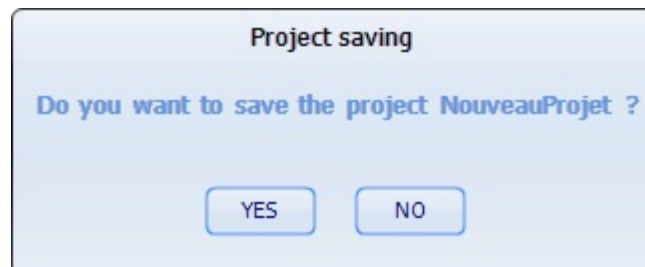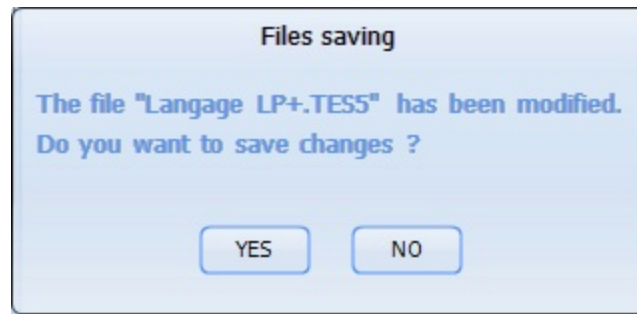
NOTE :

Contrarily to saving, closing a project allows you to review its situational analysis before its closure, reporting you any modifications made to it.

3.2 - Files Editor

Loading a tests document (tests program or equivalence table) in edition mode will display it in Winpass work zone using a text editor.

This editor (Specific to Winpass) includes its own **tools to help writing a document** and has the specificity to include a **syntactic analyzer (or parser)**. This editor type is also called a syntax highlighting editor.

CONVENTION :

A **syntax highlighting editor** analyzes phrases from a text and manage conditional rules allowing it to choose color and font styles of specific key words inside a document.

This editor type is widely used by software developers as it brings a greater readability by differentiating language elements inside their project between families or categories (Procedures, instructions, variables, data types, …).

It is now important to be able to differentiate the three main characteristics of a language :

- Its **spelling** lists every key-word recognized in this language.

- Its **grammar** gathers every operating rules of key-words in the language.

- Its **phrases** describe the way to use the language grammatical rules.

In our case, phrases from a text are **software language instructions**.

IMPORTANT :

A **syntax highlighting editor**, as every other text editor such as Microsoft Word, does not process phrases meaning (Instructions in our case).

It only **indicates whether the language's grammatical rules are observed**, while differentiating some test elements using colors.

For instance, "Oranges will always be blue" is a correct phrase in a grammatical standpoint (Spelling is also respected if you consider it to be a text in English), but, let's be honest, this phrase holds no meaning outside of this context.

3.2.1 - Editor's Properties

Winpass 5000's text editor allows you to write test programs but also equivalence tables. It is a Unicode syntax highlighting editor (meaning that it handles non-western fonts such as Cyrillic, Asian ideograms, …).

3.2.1.1 - Position markers or Bookmarks

Left clicking in the column showing the line number (1st column on the left) allows you to put a bookmark on this line. These bookmarks (9 maximum) will let you, via the editor's contextual menu, go directly to the line bearing the bookmark without scrolling through the whole document :
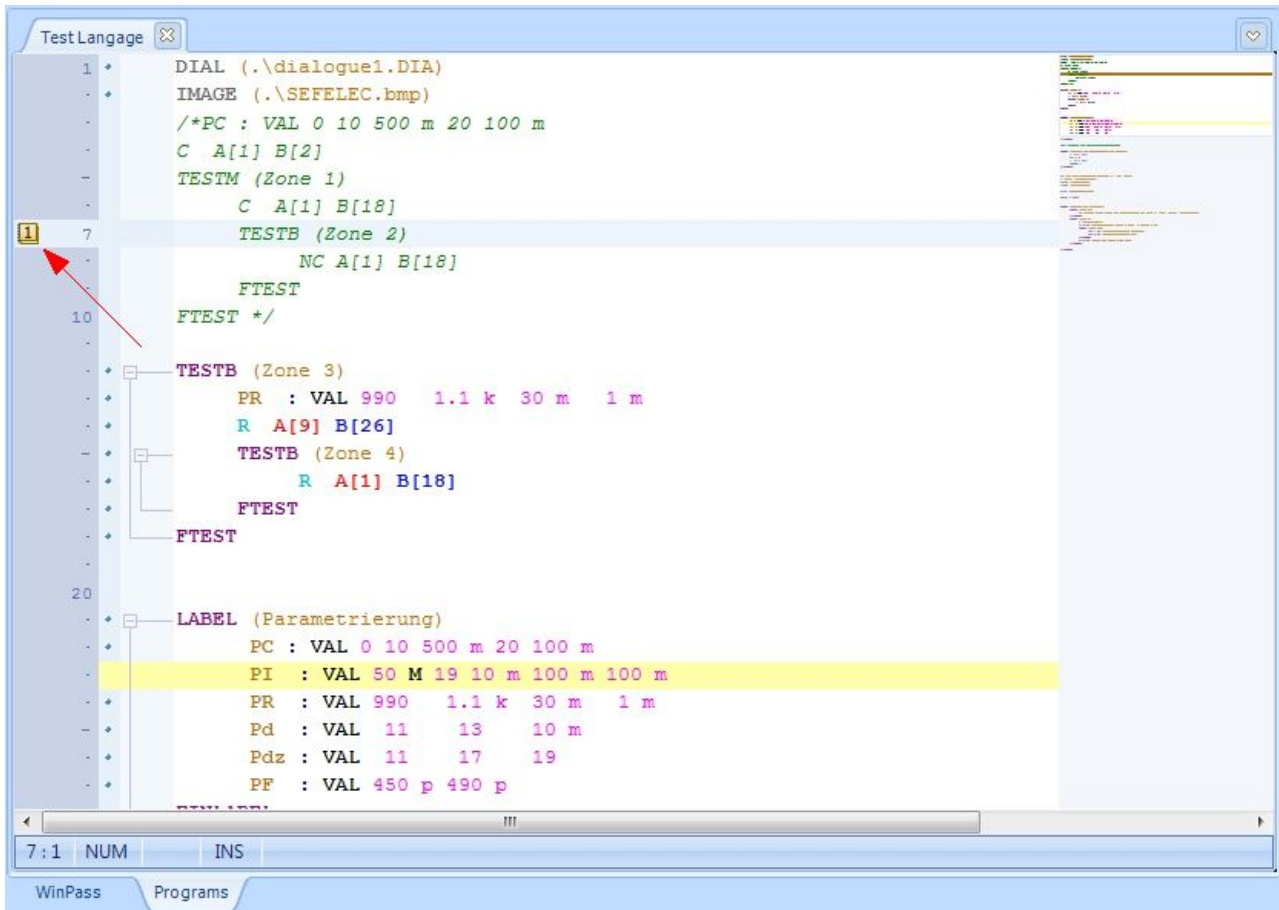
Up to this day, relocation zones handled by the editor are of types LABEL, LOT, SER, TESTB, TESTM, STIMULI, if, elseif, else, for, while (please refer to the developer's manual for more information about these instructions).

3.2.1.2 - Modification Indicator

Every modification made to a document is reported by a yellow border next to the line number. This border becomes green when saving the document :
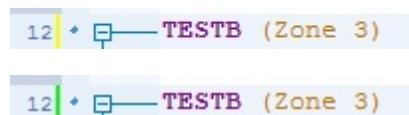
3.2.1.3 - Automated Indentation and Intelligent Tabulation

For a clearer formatting of your documents, the editor contains a tabulator of variable positions. This means that whenever you press the "TAB" button on your keyboard, the editors places the cursor on the end of a word (Separation character) of the previous line, which will align every instruction of the same block :

3.2.1.4 - Line Status Indicator

A column (to the right of line numbers) signals, using a colored point, lines checked and valid (Blue points), lines with a warning (Yellow points or background), as well as lines with syntactic or compilation errors (Purple points or background) :

3.2.1.5 - Code Folding and Instructions Blocks

This editor is not limited in document size; it contains a code folding manager allowing you to hide/display specific blocks of your document (Left click on the + or - sign) for a quicker navigation in big files. The column dedicated to code folding shows you every imbrication between zones as a tree structure or folding levels :
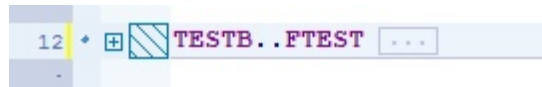
Up to this day, code folding handled by this editor are of types LABEL, LOT, SER, TESTB, TESTM, STIMULI, if, elseif, else, for, while (please refer to the developer's manual for more information about these instructions).

3.2.1.6 - Smart Scrollbar

Located on the right side of the editor, the vertical scroll bar is replaced by a minimized representation of your document. This screen allows you to navigate more easily and precisely than with a standard scroll bar, with big documents in particular :
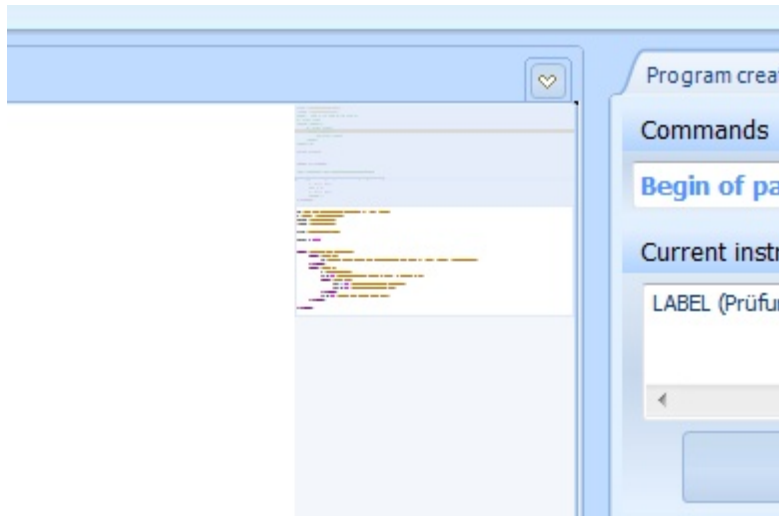
3.2.2 - Editor's Tools (Contextual Menu)

A contextual menu (accessible with a right click) gives you access to every tool from an advanced text editor but also shortcuts to standard Winpass actions.

3.2.2.1 - Undo, Redo

These two buttons allow you to come back to a previous modification state (Undo), or execute once more modification you made on your document (Redo).

↩	Undo	Ctrl+Z
	Redo	Maj+Ctrl+Z
✂	Cut	Ctrl+X
📋	Copy	Ctrl+C
📋	Paste	Ctrl+V
✗	Delete	Ctrl+Suppr
	Select all	Ctrl+A
🔍	Find...	Ctrl+F
	Replace...	Ctrl+R
	Previous	F6
	Next	F7
	Go to Bookmark	▶
	Options	▶
	Compile	
💾	Save changes	
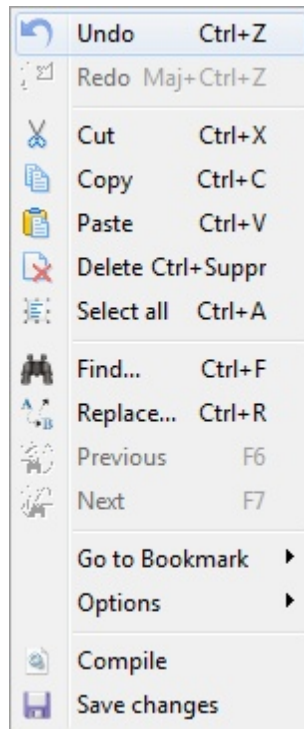
3.2.2.2 - Cut, Copy, Paste

These three buttons allow you to manipulate the selected text bloc.

Reminder : You can select a text block (Dark blue background) by holding your left mouse button or holding the Shift keyboard button.
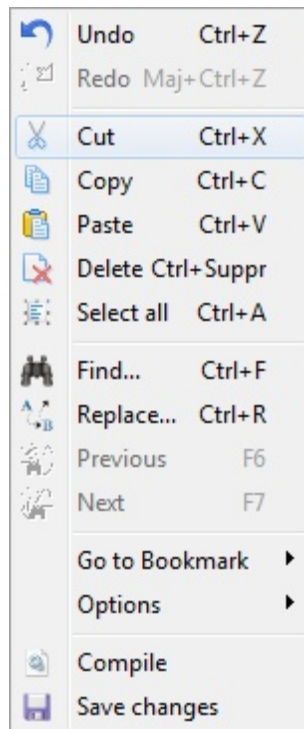
3.2.2.3 - Find, Replace

The first two buttons grant you access to text research and replacement options in all or part of your documents.

The next two buttons allow you to navigate in the current document in order to process the selected research/replacement operation.

	Undo	Ctrl+Z
	Redo	Maj+Ctrl+Z
	Cut	Ctrl+X
	Copy	Ctrl+C
	Paste	Ctrl+V
	Delete	Ctrl+Suppr
	Select all	Ctrl+A
	Find...	Ctrl+F
	Replace...	Ctrl+R
	Previous	F6
	Next	F7
	Go to Bookmark	▶
	Options	▶
	Compile	
	Save changes	

Find...

Find : A

Options
- ☐ Case sensitive
- ☐ Whole words

Direction
- ◉ Forward
- ○ Backward

Scope
- ◉ Whole document
- ○ Selected Only

Starting
- ○ Scope
- ◉ From caret

[OK] [Cancel]

Replace...

Find : A

Replace with : B

Options
- ☐ Case sensitive
- ☐ Whole words

Direction
- ◉ Forward
- ○ Backward

Scope
- ◉ Whole document
- ○ Selected Only

Starting
- ○ Scope
- ◉ From caret

[OK] [Cancel]

> **REMARQUE :**
>
> The status bar (Located at the bottom of the editor) permits you to undertake quick researches in the entire document currently edited; The only research option is case sensitivity (Upper cases, lower cases) :
>
> Aa Chercher… ▼ ▲

3.2.2.4 - Bookmarks Management

The "Go to bookmark" button allows you to align the editor directly on the lign bearing the chosen bookmark (From 1 to 9) when there is at least one active marker.

↶	Undo	Ctrl+Z
☑	Redo	Maj+Ctrl+Z
✂	Cut	Ctrl+X
📋	Copy	Ctrl+C
📋	Paste	Ctrl+V
✗	Delete	Ctrl+Suppr
≣	Select all	Ctrl+A
🔍	Find...	Ctrl+F
A→B	Replace...	Ctrl+R
	Previous	F6
	Next	F7
	Go to Bookmark	▶
	Options	▶
	Compile	
💾	Save changes	
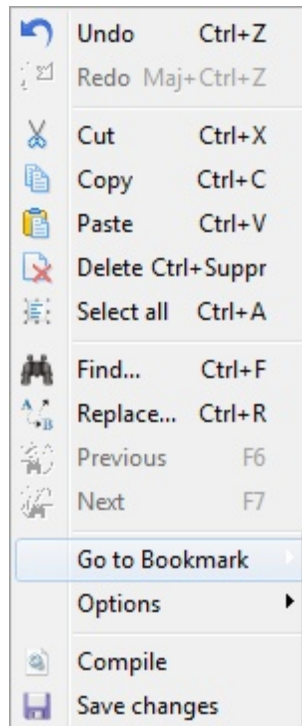
3.2.2.5 - Activating, Deactivating Editor Options

The "Options" button allows you to deactivate specific editor options (Syntax highlight, code folding, smart scrollbar) or reactivate them. Activating or not such options is kept from a Winpass execution to another.

↩	Undo	Ctrl+Z
↷	Redo	Maj+Ctrl+Z
✂	Cut	Ctrl+X
⧉	Copy	Ctrl+C
📋	Paste	Ctrl+V
☒	Delete	Ctrl+Suppr
⠿	Select all	Ctrl+A
🔍	Find...	Ctrl+F
ᴬ⁄ᴮ	Replace...	Ctrl+R
	Previous	F6
	Next	F7
	Go to Bookmark	▶
	Options	▶
⎙	Compile	
💾	Save changes	
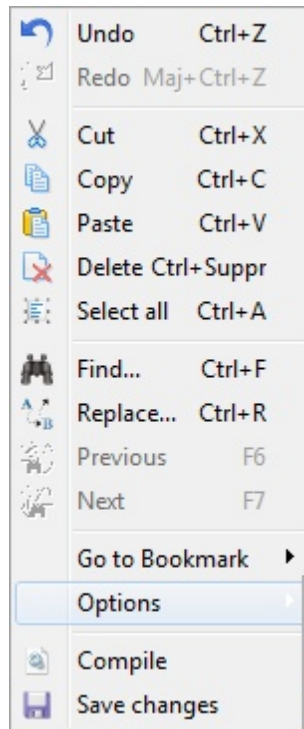
3.2.2.6 - Compiling, Saving Modifications

These two last buttons are simply shortcuts to two tasks associated to the "Edit" tab :

↩	Undo	Ctrl+Z
↷	Redo	Maj+Ctrl+Z
✂	Cut	Ctrl+X
⧉	Copy	Ctrl+C
📋	Paste	Ctrl+V
☒	Delete	Ctrl+Suppr
⠿	Select all	Ctrl+A
🔍	Find...	Ctrl+F
ᴬ⁄ᴮ	Replace...	Ctrl+R
	Previous	F6
	Next	F7
	Go to Bookmark	▶
	Options	▶
⎙	Compile	
💾	Save changes	
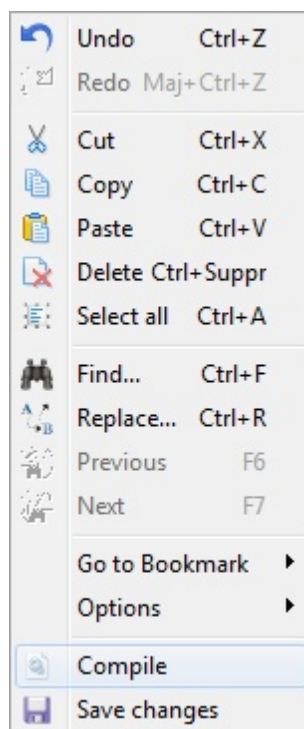
3.3 - Editing a Test Program

A test program is a text file regrouping every test and measure necessary for a product's or sub-product's validation. You might consider necessary to write several programs in order to validate your products or products line, depending on your internal quality process.

In addition to tests and measures, Winpass 5000 gives you access to an instruction set designed to format your reports, conceive conditional tests, but also manipulate values or calculate more complex expressions.

> **CONVENTION :**
>
> We could define tests and measures as every instruction allowing to make measures by questioning a Synor 5000 line tester, or every other external device recognized by Winpass 5000.
>
> **A measure consists of surveying an electronic value** mainly by injecting a current or a voltage in an element linked to a tester's matrix.
>
> **A test usually consists of performing several measures** on one or several elements linked to a tester and returning FAILED or PASS.
>
> The programming language recognized by Winpass 5000 is the **LP language** (or Langage de Programmation).This LP language mostly inherits from the one used on the older Synor 4200 line.
>
> It is expanded with Winpass 5000 specific programming instructions permitting variable, conditional tests functions and sequence loops management.

> **IMPORTANT :**
>
> The instruction set recognized by Winpass 5000 when editing a test program depends on several requirements relating to the current context :
>
> - The **tester configuration** to which is **linked** the test program inside the test project.
> - Failing that, the **tester configuration** to which Winpass 5000 is connected while editing.
> - **Measurement cards types (BT,HT-DC, HT-AC)** of the current tester configuration.
> - **Available tester** options (API, Stimuli, …).
> - **Referenced external devices** working with a set tester.
>
> When editing a test program, if no tester configuration is available, Winpass 5000 will allow the complete LP instruction set.

3.3.1 - Syntaxic Rules

An instruction always starts with a key word without space. It can then be followed by parameters separated by spaces when the instruction requires it.

Even if it is possible to put several instructions on the same line, **it is strongly recommanded to have only one per line** for better readability.

Measurements and tests instructions always have to be preceded by the related configuration instructions; But only one configuration instruction can be enough to perform a set of tests from the same type.

Most of the instructions have to be described on a single line, some tests like groups tests and equipotentials tests can be described on several lines using an end instruction delimiter.

> **NOTE :**
>
> Winpass 5000 manages **single line commentaries or block commentaries** :
>
> - A **"//"** signals the start of a commentary that will end with a simple line break.

- A **"/*"** signals the start of a multi-line commentary block that will end with **"*/"**.

Commentaries are always displayed in green and nothing in commentary will be processed by the syntax analysis or the compilation.

```
/* This
   is
   a
   multi-line
   commentary
*/

LABEL (Durchgangsprüfung)
       C A(Z001-Con1.5) B[22]
       C  A[9] B[26]
       C  A[1] B[18]              // This is a single line commentary
       C  A[2] B[18]
```

Using commentaries is part of a developer's good behavior in general. Don't be shy about using them to explain the different parts of your test programs.

3.3.2 - Help when Creating Programs

The help for test programs creation (Located in the right side of the editor) will help you familiarize yourself to the LP instruction set available for your project.

3.3.2.1 - Inserting a New Instruction

It permits you, in addition to the instruction choice, to have access to every related parameters in order to avoid syntax mistakes.

Clicking on the "Add to program" button will automatically insert the instruction at the cursor's location if the line is empty, or at the next line.

3.3.2.2 - Modifying an Existing Instruction

It will also allow you to modify instructions already present in your test program. Indeed, double clicking on an instruction line in the editor will put this instruction in the help panel. You will then be able to modify and replace it inside your program.
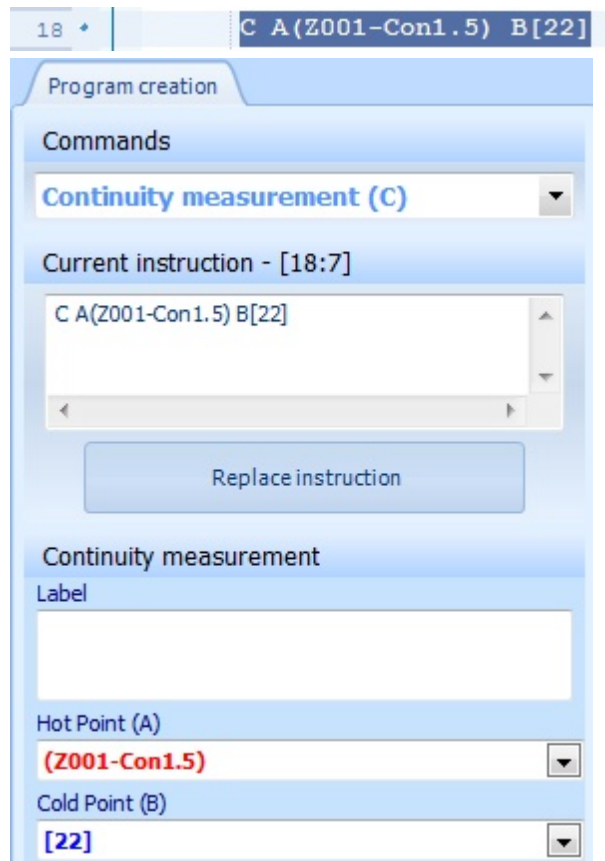
Clicking on the "Replace instruction" button will automatically update the instruction with the new modified parameters.

3.4 - Creating an Equivalence Table

An equivalence table is a text file listing every test point of a tester configuration that you judge necessary for Winpass to write down as names instead of numerical values. This equivalence concept is called "Transcoding".

An equivalence table alone has no test utility. It only exists to be paired with at least a test program in a test project. In other words, even if a transcoding has been defined in the table, nothing forces you to use it in your programs. It is optional and has no influence on the tests.

If an equivalence table is related to a program, you will gain access to this transcoding list when editing your test program in order to configure your tests and measurements more easily.

CONVENTION :
A test point is a whole number designating a point's rank in the commutation matrix inside of a tester's architecture.
A transcoding is a Unicode character string of "Equivalence" made to design a test point.

Reminder; A point from a tester's matrix can be commuted, either on the "Hot" bus, or on the "Cold" bus; A transcoding is by no mean linked to a tester's measuring bus, it is only its usage in a test program that will identify the way it is used (Hot or Cold).

IMPORTANT :
There cannot be two transcoding for a test point in the same equivalence table.
There cannot be two test points for a transcoding in the same equivalence table.

NOTE :

Transcodings described in an equivalence table are also used while executing a test program (using the equivalence table) when the generating tests report.

Indeed, test instructions described with test points (Ranks numerical values) will produce test results with the related transcoding if they exist in the equivalence table.

3.4.1 - Syntaxic Rules

Equivalence tables only describe transcoding that you want to use in your programs.

For more clarity, Winpass 5000 forces you to describe only one transcoding per line.

A tester's matrix point (Or test point) is described between brackets ('[' & ']'), a transcoding is the character string described between parenthesis ('(' & ')'). The following two syntax are recognized :

<div align="center">

(*Transcoding*)**=[***Tester rank***]**   **or**   **[***Tester rank***]=(***Transcoding***)**

</div>

Outside of transcodings themselves, spaces have no influence and are automatically filtered.

IMPORTANT :
Transcodings are case sensitive (Upper case, Lower case).
Transcoding handles character strings containing spaces. However, spaces at the start and at the end of the string are filtered.
Thus, the following transcodings are equal :
| (borne.12) | (borne.12) | (borne.12) |
When these are not :
| (borne.12) | (Borne.12) | (BORNE.12) |
| (bor ne.12) | (borne .12) | (borne. 12) |

NOTE :

Just like tests programs, equivalence tables handle **comments lines or comments blocks :**

    - A **"//"** signals the start of a commentary that will end with a simple line break.

    - A **"/*"** signals the start of a multi-line commentary block that will end with **"*/"**.

3.4.2 - Help when Creating Equivalence Table

The help tool for creating equivalence tables (located to the right of the editor) will greatly help you generate entire transcoding lists but also let you have an overview of your equivalence tables.

3.4.2.1 - Equivalence Table Properties

The "Transcoding" tab of the help panel will help you list every valid transcoding from your equivalence table;
The following table will permit you to sort validated transcodings after every equivalence table's compilation.

The "Transcoding delimiters" allow you to manage delimiters common to every transcoding from your table. After modifying at least one delimiter, if you press on the validation button (Green tick), Winpass will replace the old delimiter(s) by by the new one(s) by reconstruction every transcoding from the table.
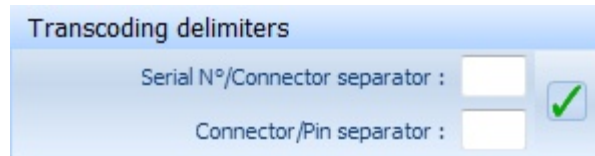
3.4.2.2 - Constructing a Transcoding List

The "Connectors" tool will allow you to construct (Semi-automaticaly) entire transcoding lists, either from standard connector types from the marker, or by defining yourself a mask of automated creation of transcoding names.

The "Connector Library" panel allows you, after selecting a referenced connector's type, to choose the connector's pins that are really used by the products you need to test.
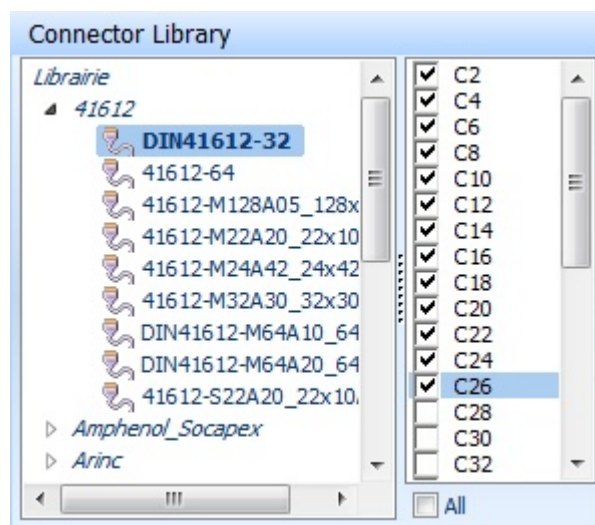
The "Transcoded point building" panel will help you define properties of the transcoding list you want to create.

Transcoded point building

Transcoding properties

Connector	Pin	#Point

Add to equivalence table

Pressing on the "Transcoding properties" opens a configuration window for the names of the transcodings that will be generated. Options are different whether you chose a connector type or not :

The related tab lists the generated transcodings depending on the transcoding properties that you validated.

Connector	Pin	#Point
demo	C2	1
demo	C4	2
demo	C6	3
demo	C8	4
demo	C10	5
demo	C12	6
demo	C14	7
demo	C16	8
demo	C18	9
demo	C20	10
demo	C22	11
demo	C24	12

The "#Point" column allows you to choose the test point for every transcoding from the list.

> **IMPORTANT :**
> **Only transcodings allocated to a test point are considered valid.**
> When editing an equivalence table and while **Winpass is connected to a physical tester**, a **double click on a cell in the "#Point" column** will allow you to use the tester **touch probe** in order to define the test point linked to every transcoding.

Once your transcoding list is defined, pressing on the "Add to equivalence table" button will insert every valid transcodings to the equivalence table currently edited and located at the mouse location in the editor.
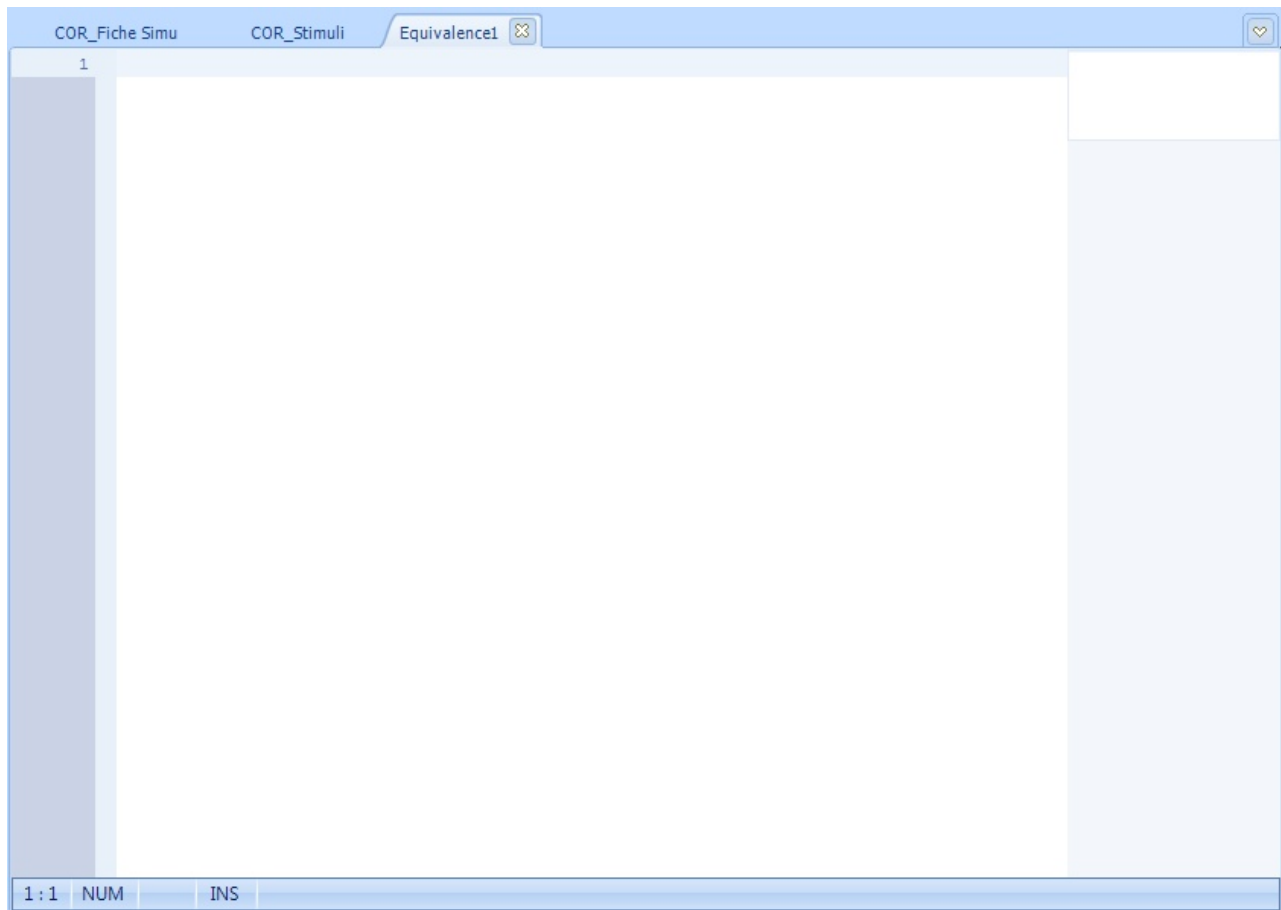
COR_Fiche Simu	COR_Stimuli	Equivalence1 ⊗	
1			

`1:1` `NUM` `INS`

Transcoded point building

Transcoding properties

Connector	Pin	#Point
demo	C2	1
demo	C4	2
demo	C6	3
demo	C8	4
demo	C10	5
demo	C12	6
demo	C14	7
demo	C16	8
demo	C18	9
demo	C20	10
demo	C22	11
demo	C24	12

Add to equivalence table

```
  COR_Fiche Simu        COR_Stimuli      Equivalence1 ⊠
     1   (demoC2)=[1]
     ·   (demoC4)=[2]
     ·   (demoC6)=[3]
     ·   (demoC8)=[4]
     ·   (demoC10)=[5]
     ·   (demoC12)=[6]
     ·   (demoC14)=[7]
     ·   (demoC16)=[8]
     ·   (demoC18)=[9]
    10   (demoC20)=[10]
     ·   (demoC22)=[11]
     ·   (demoC24)=[12]
    13

  13:1   NUM        INS
```
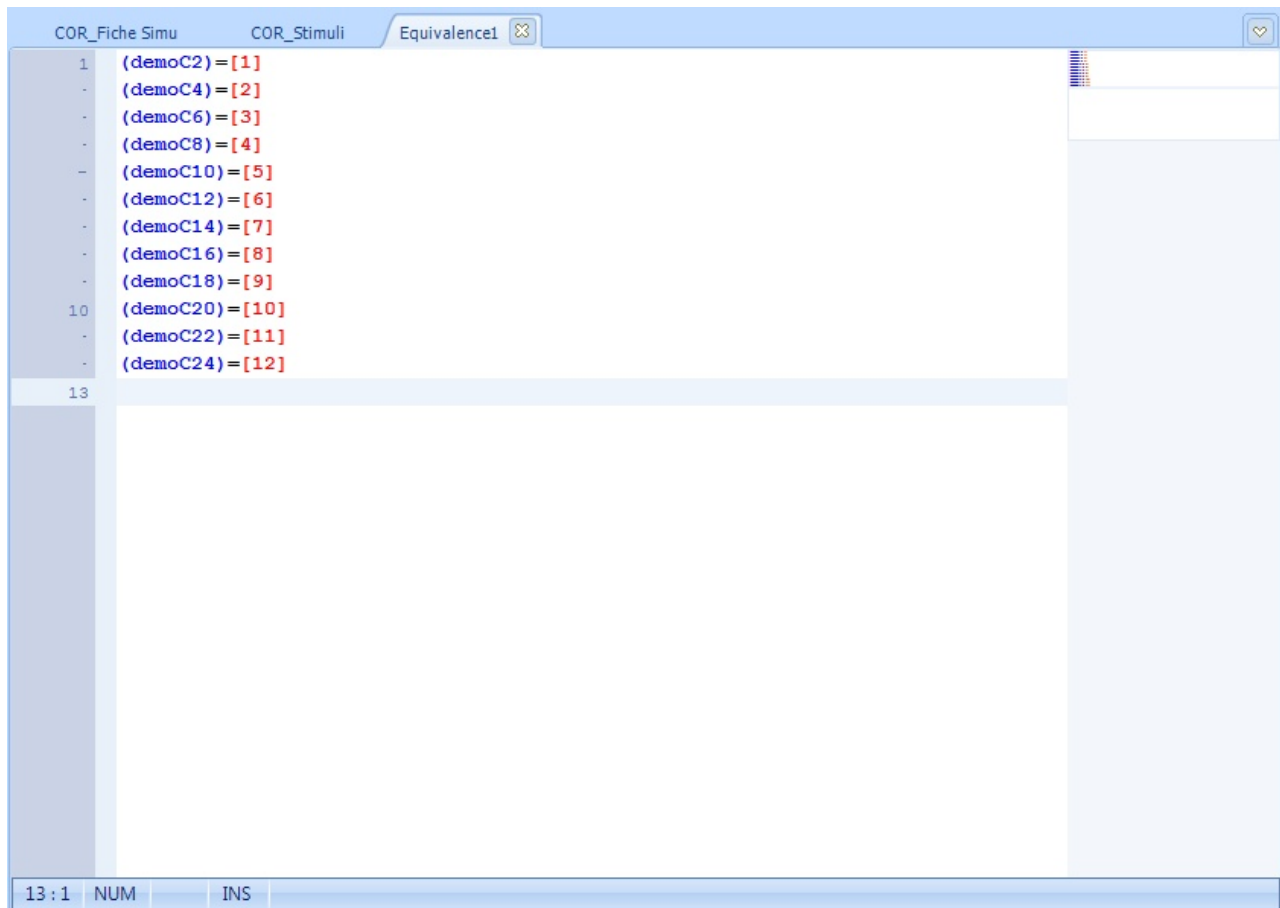
NOTE :

Inserting a transcoding list in the editor ("Add to equivalence table" button) will automatically lead to the compilation of the equivalence table which will allow you to now right away if the equivalence table is valid or if there are duplicated inside your lines of transcodings.

3.5 - Managing and Verifying Files

Every modification made to a project's test documents (test programs and equivalence tables) has to be saved but more importantly validated before using them in execution.

Validating test documents is strongly linked to the tester which they have been developed for. This implies that a test program dedicated to a tester will not necessarily be valid on another tester.

The same goes for equivalence tables listing transcodings of non-existing test points or not referring to the same type of matrix points (HT, THT, Stimuli) from a tester to another.

3.5.1 - Saving / Printing

The "Files" group from the Edit tab allows you to undertake several tasks that have no direct link to the projects.

This group permits, among other things, to create or open test documents without having to work with a project. However, keep in mind that in order to use these documents to test a product, you will have to include them in a project sooner or later.

3.5.1.1 - Saving a File

Every save done a document is final. Winpass 5000 holds no history of modifications done to a document.

Edition documents can be encrypted before saving if the option has been activated in the Winpass 5000 software configuration. In every case, documents saved by Winpass are only accessible in "read only" from every other software outside of Winpass.

3.5.1.2 - Printing a File

Documents currently edited can be printed from Winpass 5000. You can choose the printer and modify its printing configuration.

By default, the editor properties of syntax highlight are kept.

3.5.2 - Syntax Analysis / Compilation

To validate your test documents, Winpass 5000 uses two verification techniques. These are known from software developers working with advanced programming languages.
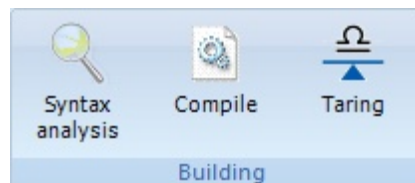
Of course, you don't need to be a software engineer to conceive a test program using Winpass 5000; and even less to create an equivalence table.

This is why Winpass implements a programming language (or LP language) based on instructions simple to understand and easy to configure. Moreover, help tools for creating test programs and equivalence tables are here to facilitate understanding the language.

3.5.2.1 - Instructions & Syntax

Programming instructions are first and foremost composed of a key-word (Also called operation code or OP code) and possibly a parameters list, which can be necessary or optional.

The entirety (OP code + parameters) defines the syntax of each instruction. Thus, it is mandatory to check first if the syntax is observed but also that instructions don't conflict with each other. This task is handled by the syntax analysis tool of Winpass 5000.

Without going into too many details, the syntax analysis allows Winpass 5000 to store internally every decoded (or decomposed) instruction that is valid.

This action is a preparatory process conditionning a document compilation; As such, you can assimilate the syntax analysis as a "pre-compilation".

3.5.2.2 - Compiler Concepts

Concerning the LP language used in test programs, you can see languages instructions as a macros realizing tasks more or less complicated that handle internally resources accessible to Winpass 5000, such as Synor 5000 testers or external devices, but also the Windows operating system or operator interactions with the screen, the keyboard and the mouse.

As for the language used in equivalence tables, it can be summarized as a single equivalence instruction translating (or transcoding) tester resources (Tests points matrix).

Considering these resources diversity, it is mandatory to check sooner or later their availability but also their good use.

This verification work is only a part of a compiler. Indeed, its main function is to convert language instructions in data compatible with the different usable resources. We are not talking here about executing an instruction but also preparing the different tasks that an instruction has to undertake.

3.5.2.3 - Analysis, Compilation, Validation

To summarize, validating a document with a syntax analysis is only conditioned by the language compliance; It depends by no mean on the context for which the document has been designed for.

However, compilation uses the document context, but also resources it requires. To do so, Winpass 5000 compiler will undertake a syntax analysis (Meaning a pre-compilation) of every instruction to determine which resources are needed for every one of them.

IMPORTANT :

The **reading context of a document** is the difference between its syntax analysis and its compilation. What is valid during a syntax analysis can be incorrect during the compilation; but the opposite is not true : a valid compilation means that the syntax analysis of a document is valid as well.

A compiler will browse a document from the start, storing what has been previously validated in the document, and can in some cases use compilation references from outside the document.

A **compilation reference** can be, either an equivalence table or a tester configuration linked to the project's element you are currently compiling.

NOTE :

In order to achieve a better understanding of concepts of context and references, and differences between syntax analysis and compilation, we can take as an example :

1- "(Borne12) = [1036]"	*(An equivalence table describing a transcoding).*
2- "PC : VAL 0 5 8.5 20 50 m"	*(A parametrization of continuity set to 8.5A under 20V max.).*
3- "C A[1] B[46]"	*(A continuity test between tester points 1 and 46).*
4- "PC : VAL 0 5 10 m 20 50 m"	*(A parametrization of continuity set to 10mA under 20V max.)*
"C A[1] B[46]"	*(Then a continuity test between tester points 1 and 46).*
5- "C A[1] B[46]"	*(A continuity test between tester points 1 and 46)*

   "PC : VAL 0 5 10 m 20 50 m" *(Then a parametrization of continuity set to 10mA under 20V max.).*

   6- "PC : VAL 0 5 10 m 20 50 m" *(A parametrization of continuity set to 10mA under 20V max.)*

   "C A(Borne12) B[46]" *(Then a continuity test between Borne12 and the tester point 46).*

Out of context (Syntax analysis), these six documents are valid as their instructions exist and their syntax is observed, but also because we are not asking Winpass to take into account the context handled by the test project which they are from.

Whereas **if we take context into account** (Compilation), validating these six documents will depend on the project structure and the interdependence rules between the language instructions :

- The 1st document is valid only if the tester configuration references at least 1036 test points.

- The 2nd document is not valid because it is impossible to parametrize a continuity measurement to 8.5A.

- The 3rd document is not valid as no parameter has been set for the continuity measure.

- The 4th document is valid : continuity is set beforehand as required by the compiler.

- The 5th document is not valid as continuity is parametrized only after the measurement request.

- The 6th document is valid only if the equivalence table is linked to the project's program.

3.5.2.4 - Results and Processing Messages

Le résultat d'une analyse syntaxique et/ou d'une compilation est résumé dans la fenêtre des messages du document que vous traitez.

Lors du traitement d'un programme de tests, les messages listés dans cette fenêtre sont classés en cinq catégories :

- "SYNTAXE" : Ce type de message signifie qu'une ou plusieurs règles de syntaxe du langage ou de l'instruction ne sont pas respectées.

- "AVERTISSEMENT" : Les messages de ce type indiquent que l'instruction ne pourra être exécutée que sous certaines conditions ou que le contexte nécessaire à son exécution est indéterminé mais non bloquant.

- "REMARQUE" : Cette catégorie signale juste que l'instruction omet certains paramètres optionnels. Ces messages ne sont affichés que pour attirer votre attention sur d'éventuels oublis de votre part.

- "VALIDATION" : Les messages de ce type indiquent les raisons pour lesquelles il sera impossible d'exécuter l'instruction. Ces messages sont essentiellement dus à des références manquantes dans le projet.

- "COMPATIBILITE" : Ce type de message signifie que le contexte dans lequel vous utilisé l'instruction est incompatible avec les paramètres de celle-ci ou tout simplement que l'instruction n'est pas compatible avec le contexte prévu par le projet.

Test Langage.TES5			
	Compilation	12/09/2016 09:10:59	
(23,22)	Compilation	[89] WARNING - Out of limit value: Corrected parameter ("UISOL" = 20 V)	
	Compilation	12/09/2016 09:10:59 - Compilation terminated with 0 error(s).	
		43 instruction(s) compiled / 66 line(s) - [0.154 sec.]	
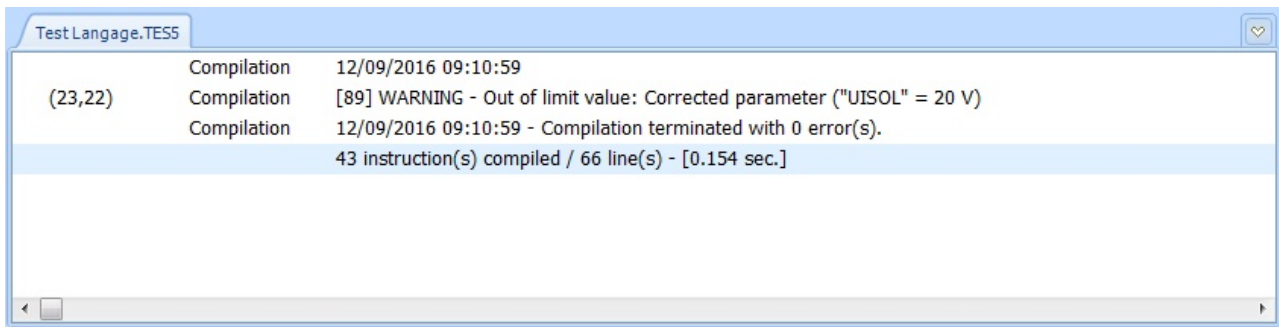
Pour une table de correspondance, Il n'existe que trois catégories de messages :

- "SYNTAXE" : Ce type de message signifie qu'une ou plusieurs règles de syntaxe de définition d'un transcodage ne sont pas respectées.

- "AVERTISSEMENT" : Les messages de ce type indiquent qu'un transcodage même valide ne pourra pas être utilisé du fait d'une redondance dans la table de correspondance.

- "ERREUR" : Cette catégorie signale que le transcodage d'un point de test n'est pas possible car incompatible avec la configuration de testeur en cours ou celle prévue dans le projet.

Test Langage.TES5	COR_Fiche Simu.COR5		
(11,1)	Compilation	WARNING - [14] Tester point described twice - [10] = (Z001-Con1.10) = (Z001-Con1.10)	
	Compilation	12/09/2016 09:13:58 - Compilation terminated with 1 error(s).	
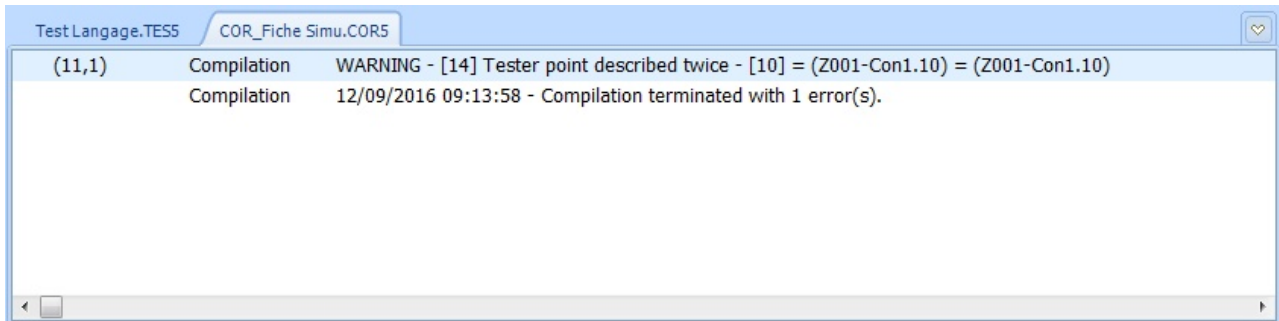
4 - Tests Execution

The execution module allows you to test and validate your products. This module is attainable only when Winpass is connected to a tester (Either Physical or Virtual).
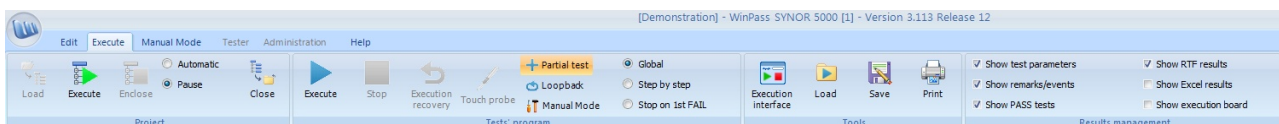
The aim of this module is to produce tests reports allowing you to validate your products. You should keep in mind though that Winpass is developed in the first place to detect critical defaults when validating a product, not attesting that a product is valid.

In other words, when you want to test that two wires are well linked together for instance, Winpass and the tester will check above all if the link is not cut.

Reminder, **the product tests are done by executing tests programs referenced in a tests project.** It is not possible to load a program alone (Outside of a project) to validate a product.

Winpass allows you to undertake tests in two ways; either by loading and executing automatically every tests program from a project, or by letting you choose one or several programs to execute.

IMPORTANT :

In order for a tests program to be executed, Winpass imposes three requisites :

     - Tests programs must be straightforward, meaning **compilable without errors.**

     - Tests instructions must be **compatible with the current** (connected) **tester.**

     - Every **document referenced** by a program must be **available and straightforward.**

Even if a test program is executable, **Winpass offers no guaranty that the executed program is appropriate for the product linked to the tester.** Choosing a test program to validate a product for which it has not been conceived can lead to a valid execution; As every computer system, Winpass does what it is asked to do :"The error factor is human most of the time !".

NOTE :

Even if a test program is partially executable, this does not mean that you can begin its execution anywhere; Winpass always starts its execution on the 1st program line. However, the programming language (LP) handles a large amount of instructions allowing you to condition the execution or non-execution of all or parts of a program.

4.1 - Executing a Project

The "Project" group of the Execute ribbon gathers several tools allowing you to test and validate your products.
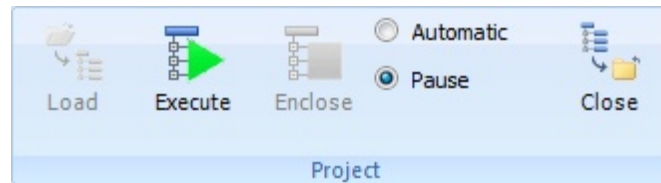
This group will permit you to load a project, but also to handle its execution without having to care about its tests programs. **Its goal is to check the different parts of a product** or to validate a group of products.

> **NOTE :**
>
> The purpose of the Winpass' tests project concept is to allow you to validate a product step by step or use several testers configurations.
>
> This operating mode has been conceived to encourage you to break down a product's test in more or less critical steps, or to isolate the different parts of a complex product requiring manual operations between the test of each one of them.
>
> The execution by project proves very useful while validating a chain of products during a test campaign that may be interrupted or postponed in time.

4.1.1 - Loading a Program

Mandatory step for every execution, loading projects will undertake several preparatory operations. Loaded projects are displayed in the project panel (To the left of the interface), which let you see that the project tree structure differs from the one in project edition mode.
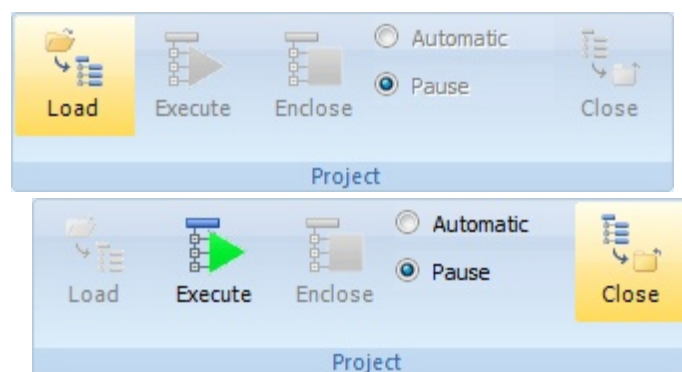
When loading a project in order to execute it, Winpass will configure it depending on the execution context :

- Only tests programs are visible in the structure (Equivalence tables are not executable files).

- Only programs referenced for a tester configuration compatible with the current (connected) tester are displayed.

- Tests programs that cannot be loaded are flagged (File not present on disk or path non-existent).

- The last execution state of every program from the project is restored (Last valid execution, invalid, partial, …).

Once the project is loaded, you can either execute it or load a tests program individually to execute it.

NOTE :

Winpass also offers the possibility to load an execution configuration allowing you to select the operating mode of the different tools linked to the execution, to authorize or forbid the use of specific tools but also to define the way test reports will be generated (See the chapter about this option for more details).

When closing a test project in execution, Winpass saves the last information about the test program's execution (Execution state, execution time, …). Thus, these information will be restored the next time the project will be loaded once again.

When reloading a project containing at least one test program that has been previously executed, Winpass displays a dialog box asking you whether you want to reset the project execution information or carry on the execution as it is.
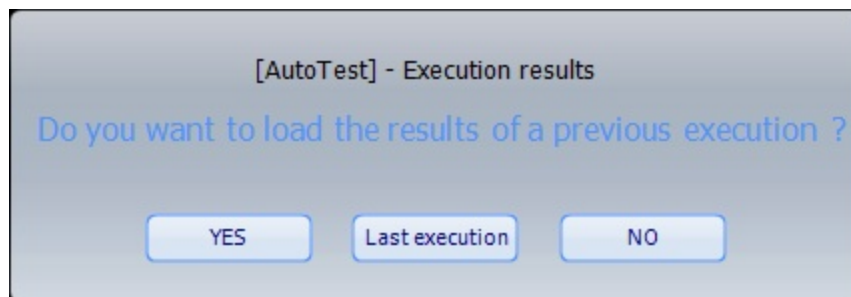
4.1.2 - Test of a Product Cluster (Range of Sub-Products)

The "Project" group controls the execution by project thanks to two buttons.

The first one allows you to start the execution of the test programs of a project, but also to resume the execution or re-execute the last FAIL test program.

The second button permits you to stop the execution of a project or to terminate the execution in project mode.

NOTE :

Stopping a project's execution leads to stopping the test program currently executed; this has for effect to flag this program to be partially executed in its test report.

Terminating an execution means that the next execution of this project will start by the 1st test program from the project structure. Terminating a project is not available while a program is currently executed.

Sequencing your tests programs inside your project is important here, as a project's execution always starts with the 1st program in the list (In the structured displayed) and continues until the last program in the list. Reminder, this sequence is customizable while in Edit mode, but not anymore in Execute.

Generally speaking, the execution in project mode automates several tasks. Winpass will handle loading tests programs, executing them then closing them once their execution is over and valid.

However, the project group lets you choose between two execution options :

    - The "Automatic" mode indicates that Winpass will execute tests program after tests program by loading/executing/closing them automatically without any external action.

    - The "Pause" mode indicates that the project's execution will be paused between each tests program's execution; whatever the execution result may be.

> **IMPORTANT :**
>
> **Continuing execution** in project mode is only available if the last test program loaded has been fully executed; this means that the prematurely stopped execution of a program (Partial test result) will pause the project's execution. You can then only re-execute it (Resume) or **stop it definitively (Terminate)**.

4.1.3 - Informations & Results

Execution results in project mode are viewable directly in the project tree structure. A pictogram is displayed next to every tests program; They let you see directly the state of the programs that have been saved since their last execution :

- ▶ : Signals that the designated program is currently executed.

- ❙❙ : Signals that the program's execution has been paused or that it has been stopped prematurely (Partial test result).

- ⚠ : Signals programs that cannot be executed, either because the file is not accessible, or following a compilation error or an error arisen during the execution.

- ✓ : Indicates that the execution result of a program (Partial or complete) is PASS.

- ✗ : Indicates that the execution result of a program (Partial or complete) is FAIL.
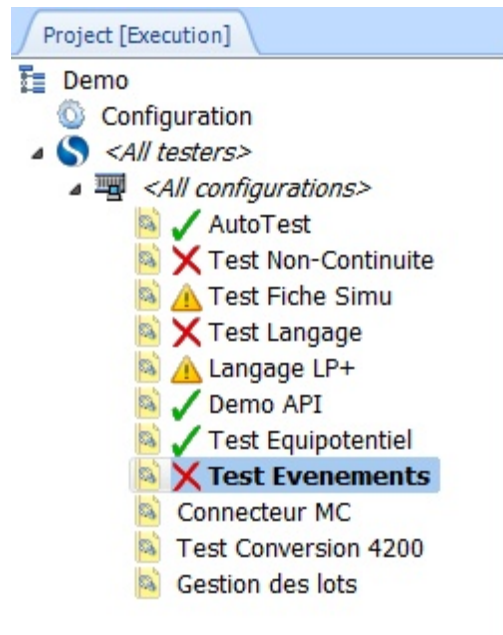
NOTE :

In "Automatic" execution mode, Winpass will not close FAIL test programs in order to keep track of their execution for analysis at the end of the project's execution.

In "Pause" mode, even if the result of a program's execution is FAIL, it will be closed.

4.2 - Executing a Tests Program

The "Tests program" group from the Execute ribbon gathers tools used to execute a specific tests program.
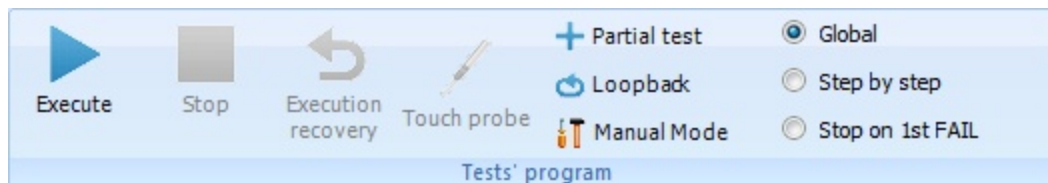
Contrarily to executing in project mode, this group will allow you to directly execute tests program and choose the way they will be executed.

However, this does not mean that this execution mode conflicts with the project mode. This last mode is only an automated extension for programs execution; **tools from the programs execution group are available while executing in project mode.**

This operating mode is mainly used when you want to test several identical products.

NOTE :

Tools and execution options from the "Tests program " group are linked and applied to the program currently displayed in the Winpass work zone.

Their use has a direct impact on the execution and the project results.

4.2.1 - Loading a Program

As said before, a test program can only be loaded and executed if it is referenced in a test project. Loading will display it in the work zone, and create a blank tests result report in a second tab.

Opening a test program is simply done by double-clicking on it in the project tree structure currently in execution.
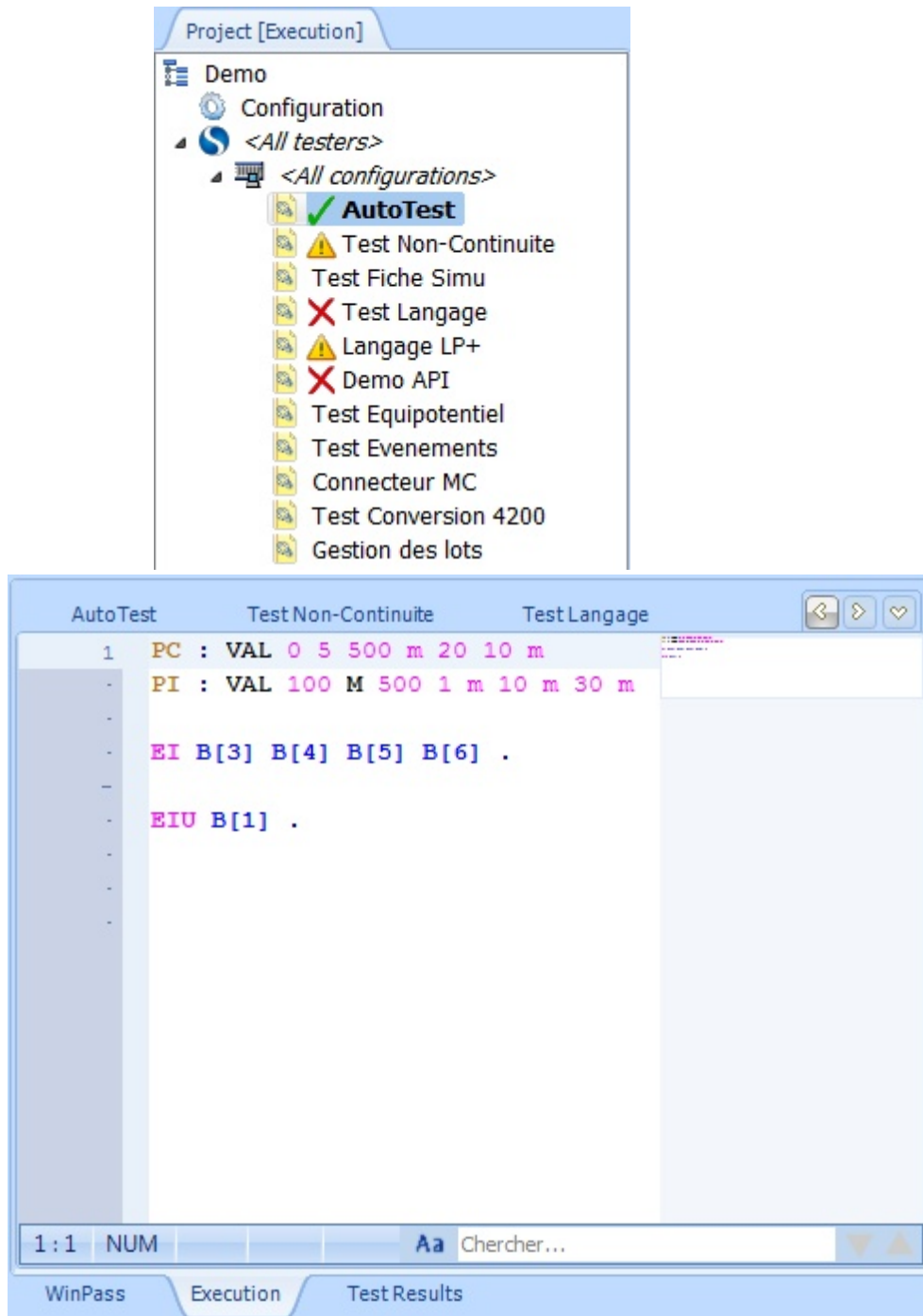
NOTE :

You can load as many programs as you desire but you can activate only one for its execution.

IMPORTANT :

Loading a test program automatically compiles it in order to check its integrity and if it is compatible with the current execution context (Connected tester).

Its execution will only be possible if this step brings no error and if every file necessary are available.

Closing/Unloading a program from the execution module is possible manually simply by clicking on the "close" cross of its tab.
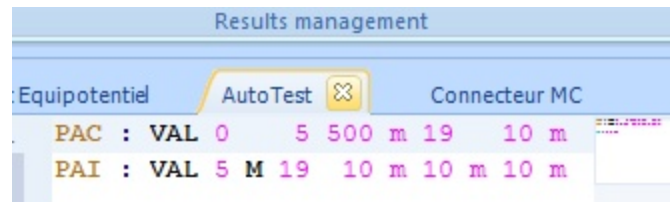
In every case, opened tests programs will be shut down once you close the tests project.

4.2.2 - Executing a Product's test

If the "Loading a Program" step is successful (Opening and compiling the file), you can apply this program's tests on your products. Two buttons similar to those in the Execution group in project mode will allow you to control your programs' execution.

A first button will let you start a program's execution, pause its execution or continue it after a pause.
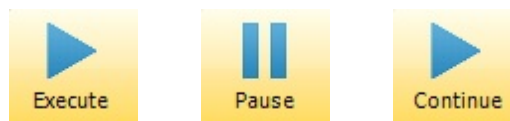
The second one lets you stop immediately the current program's execution.
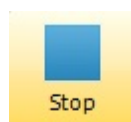
Every program's execution stopped before it finished (Partial execution) is final; such a stop will terminate the program's execution and its next execution will start over from its beginning.

IMPORTANT :

Its is important to clarify the three events that can cause an interruption in a test program's execution. Pressing the **"Pause" button** will order Winpass to stop the execution of the current instruction before returning the focus to the user. The effective realization of this action is variable and conditioned by the time necessary to execute the instruction; If, for instance, a measure is parametrized to last 60 seconds, the operator will have to wait for up to 60 seconds before having the focus back.

Pressing the **"Stop" button** will order Winpass, and the tester and/or the external devices if required, to stop as soon as possible the execution of the current instruction. This action gives you back the control almost instantly in most cases, and never exceed a few seconds in the case of tests realized with external devices.

A third interruption event is directly **linked to Winpass' internal manager of programs execution**. It is possible that an instruction cannot be correctly executed or fully executed (Losing connection, error in

the tester, parameter corrupted …), in that case and for safety concerns, Winpass can immediately stop the execution the same way the "Stop" button does.

4.2.2.1 - Test Programs Execution Options

Without going into too many details, the programs execution group possesses three tools permitting to activate options from Winpass' execution manager; these options are independents from one another and two are customizable while executing a program :

- **+ Partial test** : This button allows you to activate or deactivate the execution of specific program parts; this tool is no longer configurable while executing a program.

- **Loopback** : This tool allows you to activate the automatic re-execution of a test program when it comes to an end; a premature stop to its execution is always prioritized against automatic loopback and you can activate/deactivate it anytime during execution.

- **Manual Mode** : This button allows you to activate manual mode with stop on default when Winpass takes a measure that is outside validation criteria. This mode, which can be activated during execution, permits correcting a measurement problem or make additional measurement during the execution.

Please refer to sections of this manual describing the operating modes of these options for more information.

4.2.2.2 - Execution Sequencing Mode

The sequencing modes described here have no real interest outside from debugging and pre-test phases. They indicate to the programs execution's manager how it needs to chain a program instruction execution :

- The "**Global**" mode, which is the standard execution mode, executes instructions one after the other as long as there is no external action to pause or stop its operation.

- The "**Step by step**" mode is mainly used during validation steps for your programs; In this mode, only one instruction is executed before giving you the control back. You then have to continue manually the execution ("Continue" button) in order to execute the next instruction.

- The "**Stop on 1st FAIL**" mode allows you to secure your products tests; This mode is close to the "Global" mode, with the difference that if a test result is outside the validation bounds (Tests parameters), execution will be stopped and terminated. You can activate this mode anytime during the execution.

NOTE :

There is a variant to the "Stop on first FAIL" that permits pausing the current program's execution; The label of this mode is replaced by "**Pause on first FAIL**".

With this variant, the execution manager does not terminate the program execution and you can continue its execution manually ("Continue" button) until a possible measurement default.

4.2.3 - Informations & Results

Winpass displays in real time information about the status of the program's execution as well as its execution results.

During and after a program's execution, several levels of information are displayed in the execution interface of Winpass through several tabs and panels at your disposal :

- First the **project panel** (Left side of the interface) displays the execution state of the different programs of a project using pictogram linked to every one of them (Please refer to the project execution part for more details).

- The **execution status panel** (Right side of the interface) is dedicated to displaying information about the execution progress and will give you additional information related to the tester's status.

- The **messages window** (At the bottom of the interface) allow you to follow in-depth the execution events returned by the instructions of the test program currently executed.

- The "**Execution**" tab (In the central work zone) displaying the tests programs allow you to follow in real time the execution, line after line. This window shows no real interest outside of debugging and validating a program.

- The "**Test Results**" tab (In the central work zone) is without a doubt the most important as it is dedicated to displaying tests execution reports. It will allow you mainly to visualize in real time the test results that you defined to validate your product.
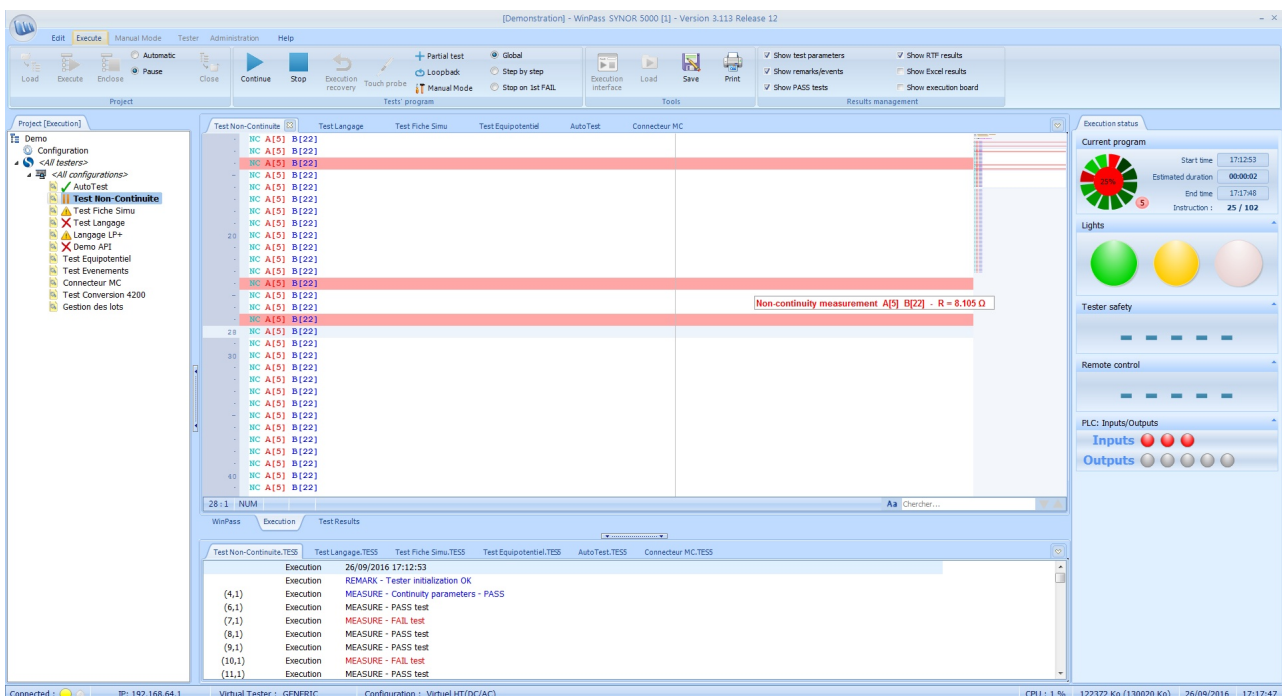
Generally speaking, Winpass defines a color code for each type of information displayed in the different panels and tabs :

- The **GREEN** indicates that an instruction, a measure or a test is sanctioned as PASS and compatible to validate a product's test.

- The **RED** is used to signal a FAIL instruction or test that is not compatible to validate the tested product.

- The **PURPLE** signals errors and execution events that prevent most of the time the pursuit of a program's execution. If however the execution makes it to the end, it will be of your responsibility to consider this type of messages as blocking or not for your product's validation.

- The **ORANGE** is used to draw your attention on sensible execution events that don't question however the validation of your products. These warnings are displayed to signal the execution of parts of program that can influence a test product's result.

- The **TURQUOISE** signals specific steps during the execution. These remarks are used to flag instructions that can potentially have an influence on the rest of the execution.

- The **BLUE** is used to give you additional execution information. These messages are mostly generated by instructions that gives more details on the execution and that you want to let appear in your reports.

CONVENTION :

The term of sanction indicates the type of result to an instruction, a test or a measurement. There are three types of sanctions in Winpass :

- The **PASS** sanction is the global result applied to tests and measurements that meet your expectations

- The **FAIL** sanction describes tests which result is outside of the validation requirements.

- The **ERROR** sanction is applied to instructions that cannot return a result.

The execution sanctions can appear in several forms in the Winpass interface :

- Directly, using the key-word form (PASS/FAIL/ERROR).
- Under the form of a text or a value displayed in color (GREEN/RED/PURPLE)
- Mixing the two forms (**PASS** / **FAIL** / **ERROR**).

IMPORTANT :

The execution of a **test program will be sanctioned as PASS** by Winpass only if every instruction, test and measurement processed are PASS.

The FAIL sanction will be applied as an execution's global result if at least <u>one</u> instruction, test or measurement have a FAIL or ERROR result.

4.3 - Tools for Monitoring Execution

The execution status panel gives you important information about the monitoring of a test program's execution. This panel is only visible when you loaded a program in order to execute it.
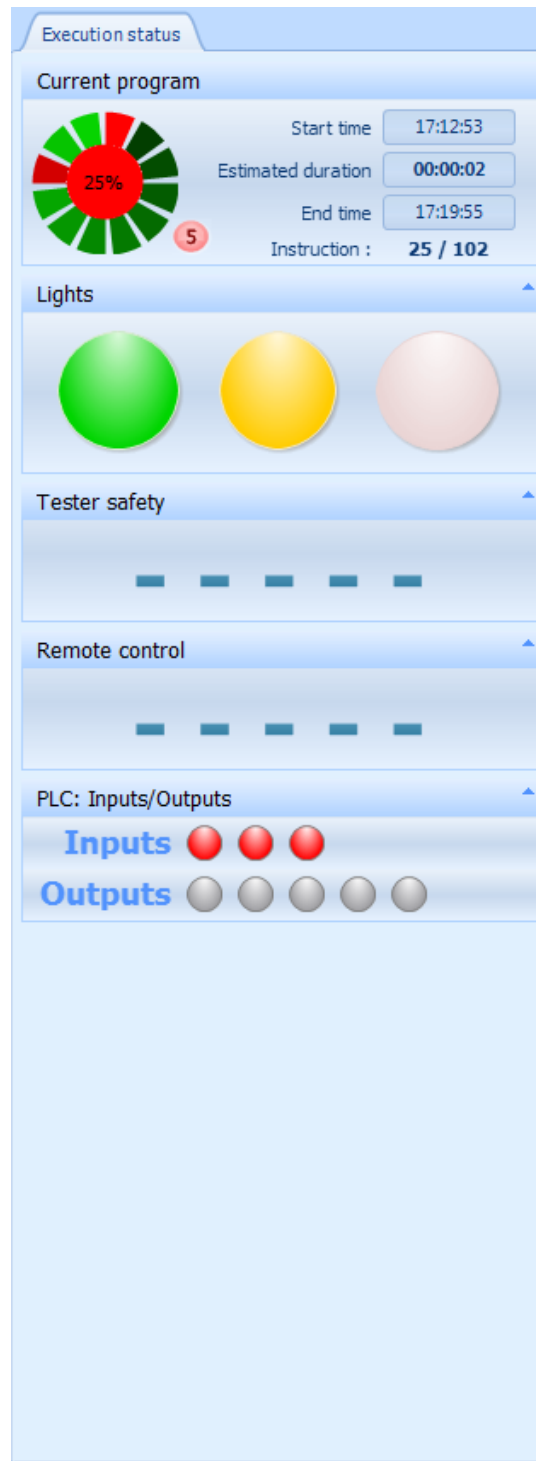
It is composed of several sub-panels that we can divide in two categories :

- The first sub-panel **"Current program"** is dedicated to information relating to the current program' s execution; meaning the one that is active in Winpass' work zone.

- The following sub-panels aim at reproducing an accurate image of **the state of options specific to Synor 5000 range testers**.

The different information displayed in these sub-panels are automatically refreshed when a state change is detected by Winpass on one or several of them.

4.3.1 - Informations about the Current Program

The "Current program" panel gives you information about the state and statistics of execution relating to the current program (Meaning the active one in the work zone).
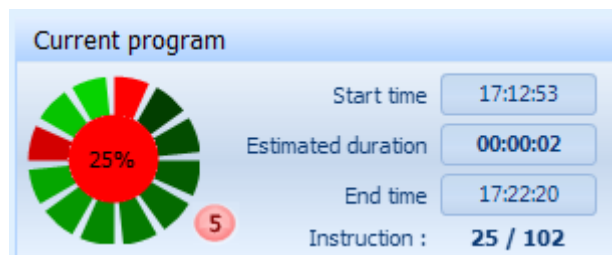
4.3.1.1 - Test validation indicator

On the right, a wheel indicator shows you the sanction of the previous twelve tests or measurements executed; a colored segment (Green or red) indicates the validation or non-validation of a test.
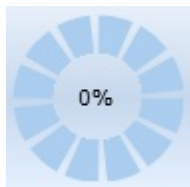
The central round of this wheel indicates the progression of the current program's execution as a percentage, but also the global result of the execution as a color, either green or red (PASS or FAIL).

A little red round (At the bottom right of the wheel) shows you the total number of tests and measurements sanctioned as FAIL.

> **NOTE :**
>
> You need to keep in mind that the **global execution result of a program is by convention always PASS at the beginning of an execution** and that if its state become FAIL during the execution, it will be kept until the end of execution.
>
> Reminder, Winpass has been conceived to detect defaults; In other words, **its goal is to give you information about the non-validation of your products**, it is up to you to analyze the tests results in order to validate or not your products.

4.3.1.2 - Execution Statistics

The right side of the panel gives you temporal information relating to the current program's execution; You can find here the execution start time, an estimation of the execution's duration and the end time.
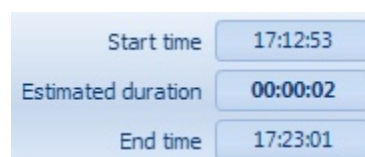
Execution's duration and end time are only statistics assessed from the previous execution time of the program. The average execution time of a program is stored in project's files referencing this program; this duration is restored each time the program is opened.

You can reset this execution duration at any time by simply clicking on it.

Finally, a last indicator displays the current instruction's number and the total number of instructions potentially executable.

Instruction :   25 / 102

4.3.1.3 - Execution Meter

In the top right, two indicators store the number of execution of the program sanctioned as PASS and FAIL using a counter in a green round and a red round.

    0   2

These two counters store only the number of program's execution since it has been loaded; closing the program lead to the loss of these numbers.

4.3.2 - Information about the Selected Tester

The execution status panel can display up to four sub-panel directly linked to the active functions of the tester to which Winpass is connected. These functions gives the possible ways of interaction between a tester and its external environment.

The document relating to Synor 5000 tester will give you more details about these options and the different ways to implement them.

Winpass only gives you a deported image of the state of these four options. However, Winpass can be brought to interact with a tester to modify the state of these functions and ask it to return their status.

> **NOTE :**
>
> When Winpass is connected to a virtual tester, it emulates the change of state of the four tester's options. You can even modify their state (under certain conditions) using the four interfaces described here.

> **NOTE :**
>
> Winpass allows you to deport the display of the four tester's options states sub-panels in separate movable and resizable windows in order to make them more visible while the operator is not in near proximity of his screen.
>
> To do so, all you have to do is double-clicking on a specific panel and resize it as you wish. To anchor it again, double-click on the deported window you wish to reintegrate to the panel.

4.3.2.1 - State of the Tester's Indicators

Testers from the Synor 5000 range use three colored indicators that let you know their operating state. These indicators are by default directly visible on the testers or deported on a light column if you acquired one.

Lights

[PHOTO voyants testeur]   [PHOTO voyants colonne lumineuse]

These three indicators are the first signs of activity and security of a tester dedicated to tests operators :

- The **ORANGE** indicator is lit when the tester is available; which means that you can connect to it using its Ethernet link. When switching on a tester, it stays turned off as long as its initialization process is not over. In other words, the ORANGE indicator always has to be lit whenever you are connected to the tester using Winpass.

- The **GREEN** indicator, when lit, signals that the tester is secured (It generates no voltage nor current) and that it is ready to execute measurements. It is off when the tester is currently used and is susceptible to generate voltage out of the commutation matrix.

- The **RED** indicator is lit while the tester is currently testing and is potentially dangerous in the security perimeter that you precisely defined beforehand; a red lit indicator is the main security information aimed at the operators working in the tester's activity perimeter. Furthermore, the red indicator is directly linked to the state of the safety loop of a tester (A red light turned off is the direct consequence of the physical opening of the safety loop).

4.3.2.2 - State of the Safety Loop

The Synor 5000 testers' safety loop is first and foremost a hardware safety commutator which, while opened, allows to forbid at any time a tester to generate high voltage in exit. Generation is then only authorized once this commutator is closed.

From the outside, it is possible to manage a software state of the safety loop by sending to the tester a check request of the real state of closure of this loop. Thus, Winpass manages a software state of closure request when needed (In particular when you start executing a test program or stop its execution).

IMPORTANT :
From the Winpass standpoint, we rather talk about **tester safety** as it handles an additional state besides the states (Opened or closed) from the physical loop.
We are talking here about a **state of request of activation (or not) of the physical loop management.**
In regards to the tester, it can only achieve tests and measurements if the physical safety loop is truly closed **AND** if a loop activation request (Software state) has been handed down to it.
In other words, if one of these two states is not valid (Opened physical loop **OR** management deactivated), **Winpass will signal you that it is impossible for it to achieve tests as long as the management is not re-activated AND the physical loop closed.**

These three states are displayed in the execution status panel like this :

Tester safety — — — — —	**The tester safety is deactivated**. This is the safety state by default. This means that the management request (Software state) of the physical security loop is deactivated and the tester cannot execute tests or measurements in high voltage.
Tester safety **CLOSED**	**The tester safety is activated and the physical loop is closed.** In other words, a safety loop management request has been sent to the tester and it has checked that it is truly closed. This is the normal working state when Winpass needs to request tests execution to the tester.
[IMAGE boucle OUVERTE]	**The tester safety is activated but the physical loop is opened.** This means that Winpass required the safety loop management to the tester but detected it opened. Every program or test execution is interrupted and prohibited as long as the tester safety is opened and the management request to close the loop has to be renewed.

4.3.2.3 - Using the Tester's Remote

It is possible to connect a remote or a pedal to a tester from the Synor 5000 range. It is not directly used by testers, its usage and the way to handle it is left free. Its sole interest is when an external software watches its state in order to undertake specific tasks.

Winpass manages a tester's remote mainly to start, interrupt or continue a test program's execution. This simply simulates a click on "Execute/Pause/Continue" button in the Test Program group of the Execute ribbon.

> **IMPORTANT :**
> **Synor 5000 testers detect remote state changes and store only two states;** the transition moment from a state to another is not saved.
> A remote or a pedal linked to a tester is then either activated or released. **Winpass is only interested in the activation state detected by the testers.**

Winpass always monitors the activation state of a remote from the moment where a test program is loaded in execution; every activation of the remote outside of this setting will not be taken into account.

In the execution status panel, only the "Activated" state is showcased as it is the only one handles by Winpass :

[IMAGE télécommande Activée]

A non-activated state "-----" means that a remote is released or that not remote is linked to the tester; it is the default state that has to be detected by a tester if no action is made on a remote.

IMPORTANT :

An "Activated" state displayed by Winpass while no remote is linked to the tester means that the electronic from the tester handling remotes is defective.

NOTE :

While executing a test program, the remote can also be used to validate the closure of dialog boxes displayed by instructions from the LP language (*See instructions Y, MB, NOTE, ECRAN, ….*).

4.3.2.4 - Activity of the PLC Input/Output Tester ports

Testers from the Synor 5000 range give you access to a port of logical signals of input/output; This port is commonly used to integrate our testers in a production chain managed by a *Programmable Logic Controller*, or PLC.

This port possesses three input signals and five output signals that can be read (Input) or positioned (Output) by Winpass using the test programs execution manager or specific LP language instructions (*See in particular instructions SE, SO, API, …*).

IMPORTANT :

Synor 5000 testers manage the PLC input/output signals depending on the HIGH and LOW logical states :

    - A HIGH level read on an input means that the tester has detected a voltage on that input.

    - A LOW level read on an input means that there is no voltage on that input.

    - An output positioned on a HIGH level means the closure of this output's contact.

    - An output positioned on a LOW level means the opening of this output's contact.

Winpass on the other hand rather uses the concept of 1 and 0 binary states (True or False) :

    - A state 1 means a HIGH level in a tester's PLC.

    - A state 0 means a LOW level in a tester's PLC.

By default, the PLC outputs of a tester are positioned on the LOW level (State 0) when starting as well as at the start and the end of a test program's execution.

In the PLC Input/Output panel, a RED round means a LOW level (State 0) and a GREEN round means a HIGH level (State 1); By default, outputs are represented by GREY rounds, which means that their state has not been modified (They are still initialized at LOW level) :
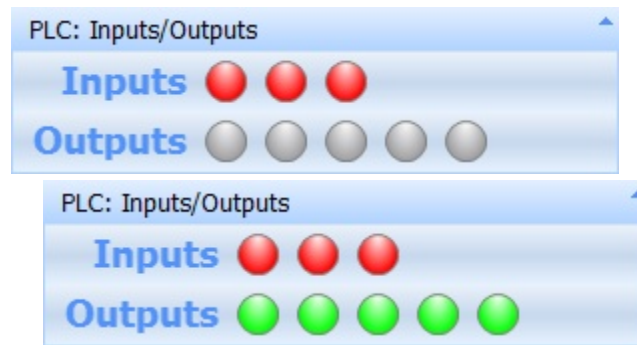
When this has been configured in Winpass, a chronogram is displayed in front of each input/output; it saves and refresh every change of state of any PLC input or output.

Winpass watches continuously the state changes of every three PLC inputs from the moment where a test program is loaded in execution; every change in the input states outside of this setup will not be taken into account.

> **NOTE :**
>
> Similar to the remote, a change of state at HIGH level of PLC input 1 or 2 from a tester are used to validate the closure of dialog boxes displayed by LP language instructions (*See instructions Y, MB, NOTE, ECRAN, ...*).

4.4 - Execution Events (Message Panel)

When a test program is loaded in execution, a messages window at the bottom of Winpass's interface is linked to it.

A first message family concerns instructions, tests and measurements that have been executed; these messages are sorted in the "Execution" category.

Finally, tests programs loaded in execution are re-compiled before every execution and instructions are controlled as well before their execution; Potential compilation problems detected are sorted in a second family named "Compilation".

> **NOTE :**
>
> The window displaying the test program execution messages saves the position of the program instructions referenced by the displayed messages; but also the position of the execution result potentially written in the test report once the program has been executed.
>
> **A simple click on a message line permits positioning the concerned program editor on the instruction line referenced by that message, and on the execution result line linked to the message.**

4.4.1 - Pre-Compiling Messages

Messages from the "Compilation" family are the result of the automatic pre-compilation (or recompilation) done when loading a test program in the Execute module. These messages are identical to those generated during the compilation of test program in the Edit module of Winpass.

Test Fiche Simu.TES5		
(51,7)	Compilation	VALIDATION - [101] Parameter PCBN : not defined
(55,7)	Compilation	VALIDATION - [101] Instruction Id forbidden in a LOW LEVEL area (BN)
(69,7)	Compilation	VALIDATION - [101] Instruction d forbidden in a LOW LEVEL area (BN)
(70,7)	Compilation	VALIDATION - [101] Instruction d forbidden in a LOW LEVEL area (BN)
(74,7)	Compilation	VALIDATION - [101] Instruction dz forbidden in a LOW LEVEL area (BN)
(75,7)	Compilation	VALIDATION - [101] Instruction dz forbidden in a LOW LEVEL area (BN)
	Compilation	27/09/2016 09:35:42 - Compilation terminated with 21 error(s).
		70 instruction(s) compiled / 86 line(s) - [0.091 sec.]

During the pre-compilation phase, equivalence tables are also checked to insure that every test point they describe is compatible with the current test. These messages may be displayed when an equivalence table linked to a test program is not usable with the test context (Tester architecture incompatible, …).

[IMAGE erreur table de correspondance]

Finally, every instruction from a program is validated whenever you need to execute it; pre-compilation messages can then appear during a program execution if the execution context has been modified by executing other instructions.

> **IMPORTANT :**
> Outside of remarks and warnings, every other pre-compilation messages forbid the execution of a tests program or can stop its execution.

4.4.2 - Execution Messages and Events

The "Execution" family lists messages generated by the execution of a test program instruction or when execution events have been returned by the tester or external devices.

These messages are mainly generated to give you detailed information on why an instruction is sanctioned as PASS or FAIL, or events that provoke a stop during a test program execution.

These events and information are sorted between five categories that respect the color code proper to the execution :

- A **MEASURE** message is displayed when a measure or a test is sanctioned as FAIL, meaning non-compliant with your standards of validation for you product.

- An **EVENT** is the consequence of a message returned by the tester or an external device that prevent the pursuit of a program execution (Test interrupted, safety loop opened, impossible measurement or undetermined result, …).

- An **EXECUTION** message is the result of the execution attempt of an instruction that failed (Incompatible execution context, tester or external device non-usable, …).

-   A **WARNING** signals primarily a specific context of execution of an instruction that can modify or influence the sequel to a program's execution.

-   A **REMARK** message is displayed to signal an important execution step that has no direct influence on the global execution result of a program (Tester initialization, test parametrization, information save during the test, ...).

```
┌─────────────────────────────────────────────────────────────────┐
│  Test Equipotentiel                                         ⌄     │
│                                                                   │
│  Tester used          : VALIDATION_MIP (N° 1003898)               │
│  Internal ID          : DEMONSTR                                  │
│  Winpass account      : Demonstration                            │
│                                                                   │
│                                                                   │
│  Test start - 27/09/2016 10:05:19                                 │
│                                                                   │
│                                                                   │
│  Continuity parameters                                            │
│       Minimum resistance              = 0 Ω                       │
│       Maximum resistance              = 5 Ω                       │
│       Test current                    = 500 mA                    │
│       Maximum voltage                 = 20 V                      │
│       Hold time                       = 10 ms                     │
│                                                                   │
│  Insulation parameters                                            │
│       Minimum resistance              = 100 MΩ                    │
│       Insulation voltage              = 500 V                     │
│       Breakdown current               = 1 mA                      │
│       Rise time                       = 10 ms                     │
│       Hold time                       = 30 ms                     │
│       Range                           = AUTO                      │
│                                                                   │
│  Continuity + Insulation test / lower  B[3] ... [6]  -  FAIL      │
│                                                                   │
│  Continuity + Insulation test with other  B[1]  -  PASS           │
│                                                                   │
│  FAIL TEST                                                        │
│                                                                   │
│  WARNING : TEST AND MEASUREMENTS IN VIRTUAL MODE !                │
│                                                                   │
│  Test end - 27/09/2016 10:05:20                                   │
│                                                                   │
├───────────────────────────────────────────────────────────────────┤
│   WinPass    │   Execution   │  Test Results  │                    │
└─────────────────────────────────────────────────────────────────┘
```

NOTE :

Winpass' philosophy being first and foremost to detect problems that might prevent validation of your products, instructions that produce results compliant with your expectations will not generate messages in the execution messages window.

It is conceived to signal remarks, defaults and execution errors; Nonetheless, valid tests and measures (Sanctioned as PASS) appear in that window for information purposes with the "PASS TEST" mention in black.

4.5 - Results Display and Management

The result of the validation test of a product is displayed in the "Test Results" tab of the work zone. This window manages different formats of test reports supported by Winpass.

In every case, tests and measures sanctioned as PASS are always displayed in **GREEN** and tests non compliant to your expectations (Sanctioned as FAIL) are displayed in **RED**.

> **IMPORTANT :**
> Reminder, instructions, tests and measurements sanctioned as PASS don't condition a product's validity. For Winpass, the presence of non-valid tests and measurements (FAIL or ERROR) in an execution report lead to the non-validation of a product.
> In other words, a **tests report validates a product only if no default has been detected during its execution.**

Whatever mode or format chosen for display, save or print these test reports, instruction, tests and measurements sanctioned as FAIL will always be present.

> **NOTE :**
>
> You cannot change the mode or display format during the execution of a test program.
>
> However, you have the possibility to change the mode or format by suspending the program's execution (Pausing the execution).
>
> That way, you can also save or print partial test reports.

4.5.1 - Filtering Execution Results

The "Results management" group of the "Execute" ribbon gives you access to three additional execution result filtering modes :
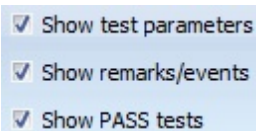
- "**Show test parameters**" allows you to include or not in your reports information regarding tests and measurements parametrization.

- "**Show remarks/events**" allows you to filter execution results to hide remarks and events that don't matter for a product's validation.

- "**Show PASS tests**" allows you to hide in your reports tests and measurements sanctioned as PASS when you only want to see defaults that would prevent validation.

A test report is complete when these three modes are activated, and Winpass will produce simplified test reports when these three modes are deactivated.

AutoTest

Tester used   : TEST (N° 1001234)
Internal ID   :
Winpass account : Demonstration

Test start - 27/09/2016 10:10:27

Continuity parameters (AutoTest)
    Minimum resistance   = 0 Ω
    Maximum resistance   = 5 Ω
    Test current   = 500 mA
    Maximum voltage   = 19 V
    Hold time   = 10 ms

Insulation parameters (AutoTest)
    Minimum resistance   = 5 MΩ
    Insulation voltage   = 19 V
    Breakdown current   = 10 mA
    Rise time   = 10 ms
    Hold time   = 10 ms

Self-Test execution (Matrix points) - PASS

PASS TEST

WARNING : TEST AND MEASUREMENTS IN VIRTUAL MODE !

Test end - 27/09/2016 10:10:30

WinPass   Execution   Test Results

AutoTest

Tester used	: TEST (N° 1001234)
Internal ID	:
Winpass account	: Demonstration

Test start - 27/09/2016 10:10:27

PASS TEST

WARNING : TEST AND MEASUREMENTS IN VIRTUAL MODE !

Test end - 27/09/2016 10:10:30

WinPass   Execution   Test Results

4.5.2 - Displaying Formats of Execution Results

Winpass allows you to display tests reports under three forms and two different files formats :

☑ Show RTF results

☐ Show Excel results
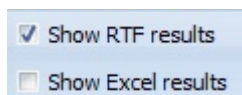
- "**Show RTF results**" will display the test reports as a text document in Microsoft's RTF format. This display format is the one that will give you the most information relating the your product's tests and is the best suited to product test reports intended for archiving.

- "**Show Excel results**" will produce tests reports in Microsoft's Excel sheets format, more adapted to analyze measurement statements. That is why this report type only contains results from the measurements realized while testing a product.

Finally, when you don't want to display your test results in neither RTF nor Excel, Winpass activates a third display mode which is a Time Line. This type of display allows to :

- Know the moment and the duration of an instruction's execution on a temporal axis.

- Gather tests and measurements on one side (Under the axis).

- Display potential remarks and events generated during the execution (Above the axis).

However, you should not consider this last display mode as a true test report, it is only meant for a quick analysis at the time of the execution.
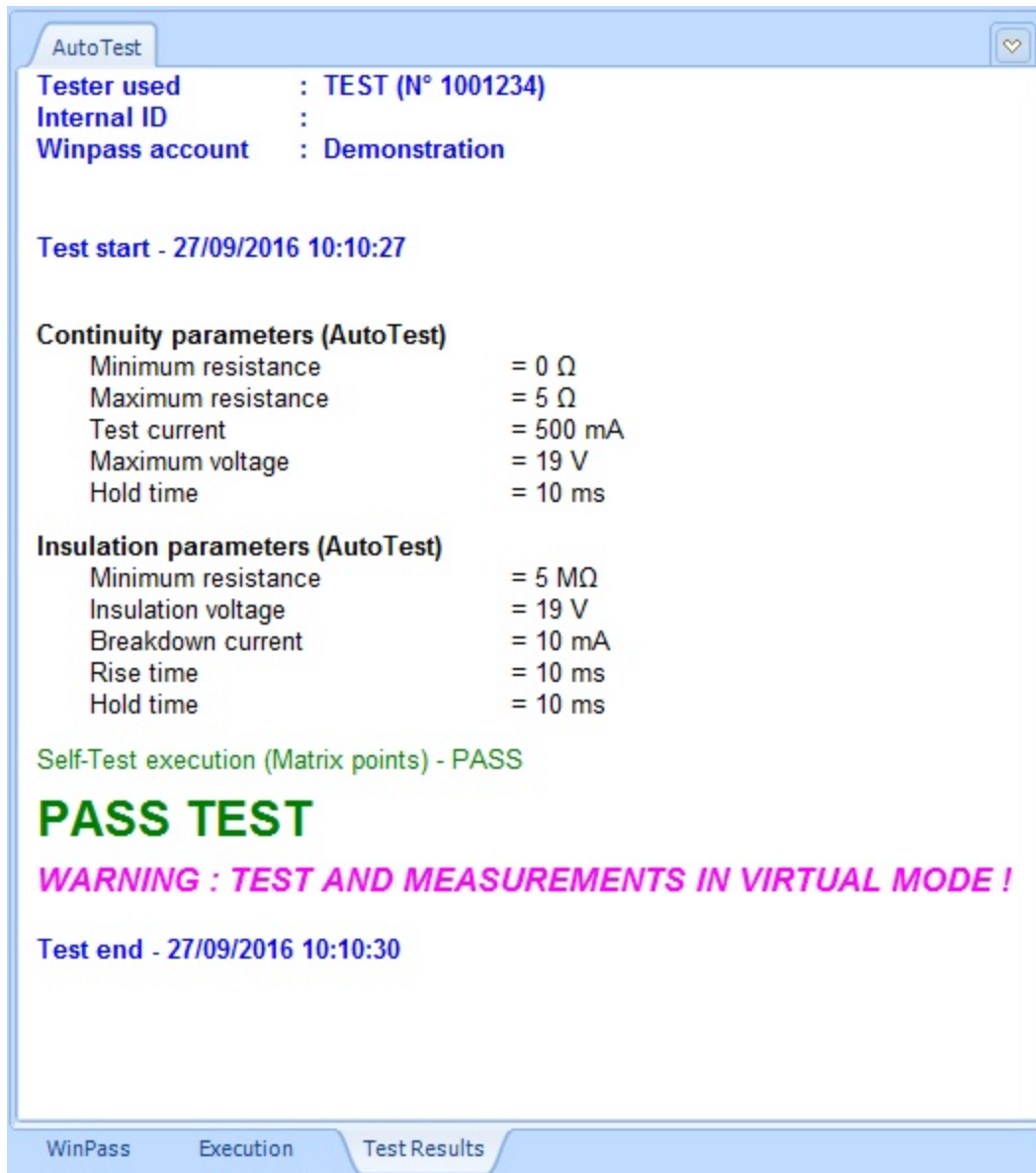
AutoTest											
	#	LABEL	Title	Tests	Details	Hot Pts	Cold Pts	Status	Results	Values	Instruction T5K
27/09/2016 10:10:27	1			Continuity parameters (AutoTest)				PASS	PASS	VIRTUAL	PAC : VAL 0 5 500 m 19 10 m
					Minimum resistance				RMIN = 0 Ω	0	
					Maximum resistance				RMAX = 5 Ω	5	
					Test current				ITEST = 500 mA	0,5	
					Maximum voltage				UMAX = 19 V	19	
					Hold time				TA = 10 ms	0,01	
27/09/2016 10:10:27	2			Insulation parameters (AutoTest)				PASS	PASS	VIRTUAL	PAI : VAL 5 M 19 10 m 10 m 10 m
					Minimum resistance				RMIN = 5 MΩ	5000000	
					Insulation voltage				UISOL = 19 V	19	
					Breakdown current				ICLAQ = 10 mA	0,01	
					Rise time				TM = 10 ms	0,01	
					Hold time				TA = 10 ms	0,01	
27/09/2016 10:10:30	4			Self-Test execution (Matrix points)		1		PASS	PASS	VIRTUAL	AT : 1 1 35

WinPass   Execution   Test Results

Name	Type	Persistent	Value (.val)	Status (.bool)	Text (.text)	Description (.desc)	
$TEST_ORDER	System		N/A	N/A		Production order number (PO)	{#0} N/A
$TEST_LOTNUMBER	System		N/A	N/A		Test lot number	{#0} N/A
$RESULT_NAME	Global		N/A	N/A	AutoTest	Result file name	{#0} N/A
$RESULT_COMMENT	Global		N/A	N/A		Execution comments/remarks	{#0} N/A
$RESULT_GLOBAL_PROG	Global		N/A	PASS	N/A	Program global result	{#0} N/A
$RESULT_GLOBAL_TEST	Global		N/A	PASS	N/A	Measurement global result	{#0} N/A
$LOT_SERIALNUMBER	Global (LOT)		0	N/A	N/A	Lot serial number	{#0} N/A

4.5.3 - Saving & Printing

By default, Winpass neither save nor print any test report following a program's execution. You will see nonetheless that using an execution configuration associated to your test project, you will be able to save or print automatically reports and even condition these productions.

In every case, the "Tools" group of the Execute ribbon allows you to save or print test reports (RTF) and measurement statements (Excel). Both of these actions are yet inaccessible during a test's execution (Pausing the execution can avoid this restriction).

Saves and prints are done on reports currently displayed; Thus; When a specific test report (RTF) or measurement statement (Excel) is displayed, both document will either be printed or saved.

When saving, a dialog box will allow you to choose your report's name and the location where you want to store it.

When printing, you have the choice of printer and you can modify its printing configuration.

Chapter
Chapitre
Kapitel

V

Advanced Execution

V - Advanced Execution

This chapter details every tool of Winpass 5000, their operation and their specific configurations features.

Some of the functions described here are optional and therefore they might not be useful to test your products.

Other functions are available only if your testers have specific commercial options or only if you subscribed to dedicated software licenses.

Finally, the context of use of Winpass being strongly user oriented, accessing some of Winpass tools can only be achieved if the current user account has been defined with rights allowing it to access said tools.

Winpass' software configuration panel has on itself a significant impact on accessing and using specific tools.

1 - Managing Users Accounts (Session)

Winpass 5000 software supplies numerous tools dedicated either to designing test projects or testing your products directly.

Every actor using Winpass does not need, in most cases, to have access to every functions of the software. That is why Winpass has been developed since its beginning to work with customizable access rights dedicated to every type of user.

To do so, **accessing Winpass is done using a sessions;** which aim to only make accessible specific functions to a user, functions which he is truly bound to use.
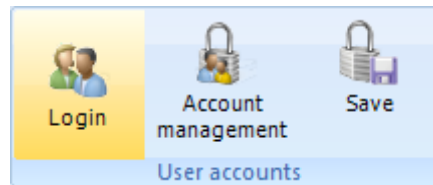
1.1 - Principles / Operating Rules

The following diagram represents the working principle of user accounts used place in Winpass 5000 :
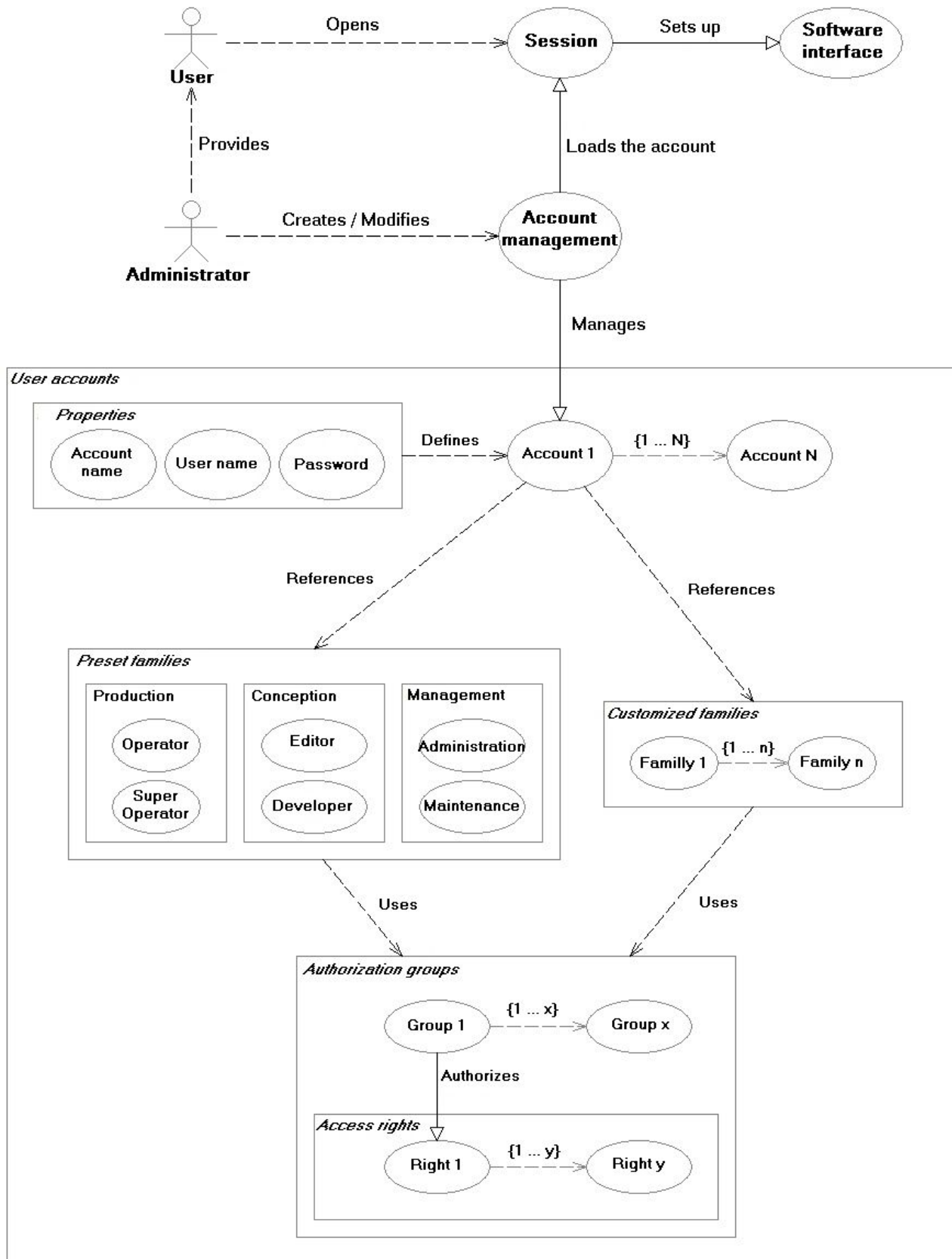
1.1.1 - User sessions and accounts

Opening a session is mandatory to use Winpass 5000. For every user susceptible to use Winpass you will have to setup an account containing in particular information required to open a session.

The session manages every defined rights for a user as long as it is open.

A user account is defined by three properties of its own : the account name, the user name and a potential password.

1.1.1.1 - Account Name

An account name is unique and indifferent to upper and lower cases used.

This identifier is used by Winpass while generating tests reports when you want to make appear the operator that made a product's test.

It is also visible at any time in the title bar of Winpass 5000.

[Demonstration] - WinPass SYNOR 5000 [1] - Version 3.113 Release 12

1.1.1.2 - User Name

As its name suggests, **the user name appoints the person or the operator type that uses Winpass.** It can be entered in both upper or lower cases during authentication.

This user name can be identical for several accounts when you want to define anonymous accounts; meaning accounts that don't name persons directly.

When you test a product, the user name is used as name for the test operator. It can appear in your tests reports if you set it up in the tests projects execution configuration.

```
Tester used        : TEST (N° 1001234)
Internal ID        :
Winpass account    : Demonstration


Test start - 28/09/2016 14:30:46
```

1.1.1.3 - Password

In order to secure accounts usage, you can link a password to every defined account.

A password is optional, but once it has been defined, its entry is case-sensitive when opening sessions.

If needed, account passwords can be modified anytime.

NOTE :

Winpass 5000 is installed with a default demonstration account in let you access the software the first time, learn the software or train operator bound to use it.

To start a "Demonstration" session, enter the user name "demo" and click on OK. This account does not require any password.

This account is bound to be deleted and replaced by other accounts more restrictive in terms of rights when you wish to use Winpass for real products testing.

1.1.2 - Accounts and Families of Rights

Users accounts manage rights, authorizations and working modes of Winpass 5000.

Opening a Winpass session with a specific account defines the scope of use of the software, defined by the account through authorization options and/or management modes.

To ease this task, **Winpass references six families of rights presets**, the most widely used, to work with the software. You can also define your own families of rights if needed.

For more information, a complete list of the defined rights contained in these families is available in annex.

1.1.3 - Authorization Groups

The several access rights and management options are sorted in authorization groups that allude directly to the different modules of management or tools groups (Tabs of ribbon) of Winpass.

The presence of a specific authorization group in a user account means that this account can access this stated module; its absence in the account means that the user is not authorized to use tools inside this group.

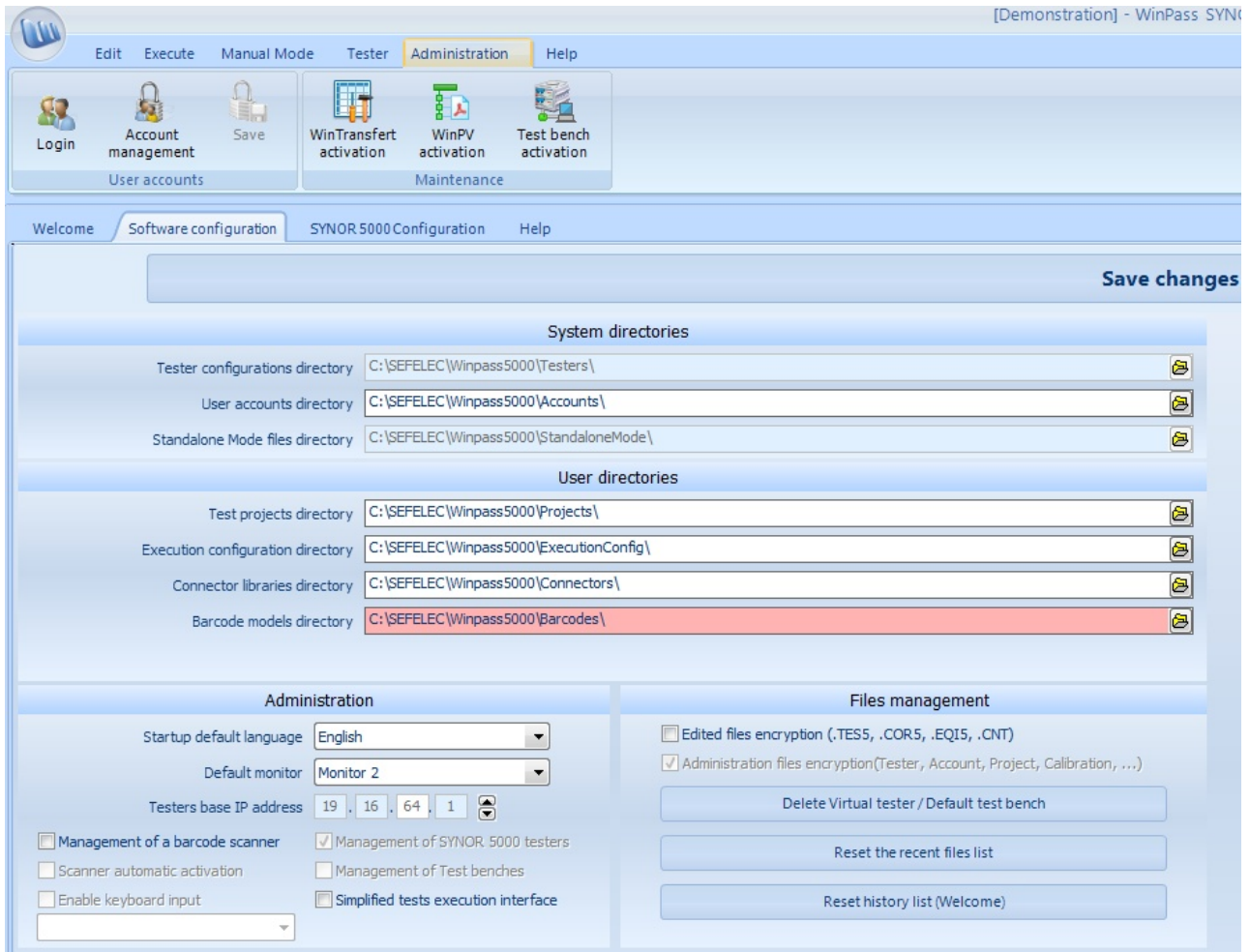## 1.1.4 - Access Rights / Management Options

Authorization groups describe every tool a user account can use or not. Deactivating a tool or a function in an account results, in most cases, to its disappearance in Winpass' interface.

Some management options described in authorization groups designate the working modes of tools from the said group. They influence Winpass' behavior when certain tools or functions are used.
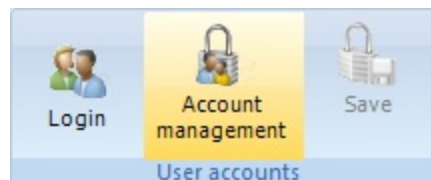
1.2 - Context / Authorizations

1.2.1 - Software Configuration

The software configuration panel of Winpass let you choose the directory where users accounts you defined will be saved.

1.2.2 - Users Rights (Session)

Configuring users accounts is only possible if the current session has the "User account management" authorization group.

This group references three account management options :

- "**Open/close a session manually**" let you switch session while using Winpass via the Administration tab ("Login" tool).

- "**Creating new accounts**" let the Winpass user create new session accounts.

-    **"Modify existing accounts"** let the Winpass user modify or delete existing user accounts.

IMPORTANT :

The account used for the current session cannot be modified nor deleted. You need to open a new Winpass session by using a different account in order to modify it.

This management aims at securing accounts that have the right to manage every user account and forbid the user to modify its own access rights defined by a third party.

1.3 - Users Accounts Management

The "User accounts" group from the Administration tab of Winpass gives you the tools necessary to manage account and session from the software.

1.3.1 - Sessions Management

The "Login" button lets you close the current session and open a new one. The standard session opening dialog box is displayed.

You need to enter an account valid information in order to open a new session. If you cancel the account switch, the previous session will stay active.

> **IMPORTANT :**
> The opening session window allows for **three attempts at account change**, then it will be automatically closed.

1.3.2 - Accounts Referencing

A simple click on the "Account management" button will display the interface of user accounts management in the work zone of Winpass. This interface split in three parts :
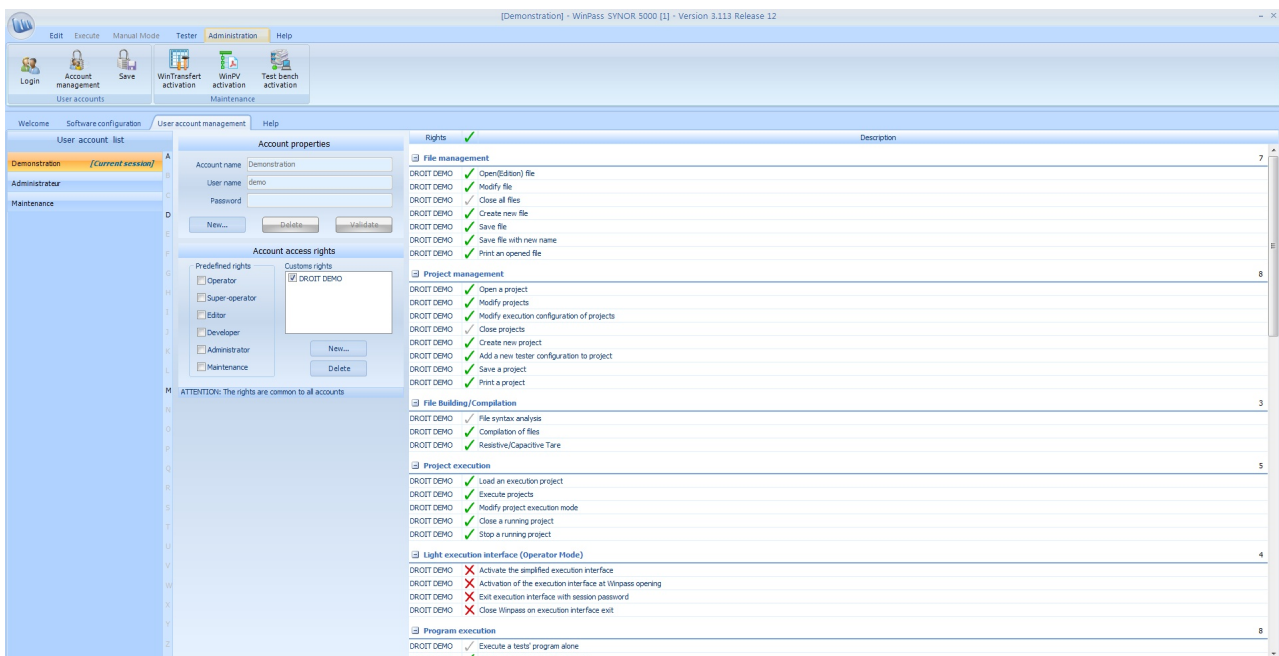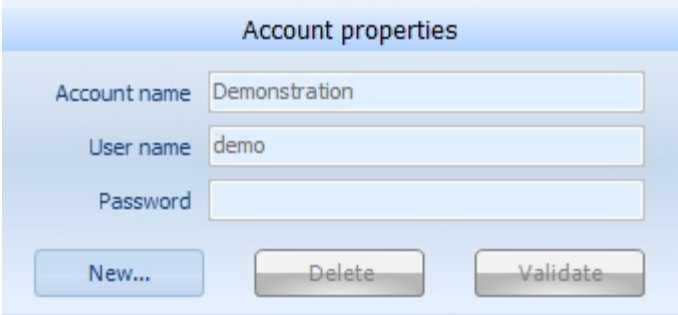

To the left are displayed every saved accounts; to the center, properties of the selected account or those of a new account; and to the right, every groups and access rights of the new account or the currently selected one.

> **IMPORTANT :**

Every modification or addition of account is not saved automatically, you need to click on the "Save" button of the ribbon so that your modifications are taken into account permanently.

1.4 - Account Properties

The "Account properties" panel let you visualize information of an existing account or enter information of a new one.

The "**New**" button let you create a new user account. You then have to type, at the minimum, an account name as well as a user name in order to be able to "**Validate**" this new account.
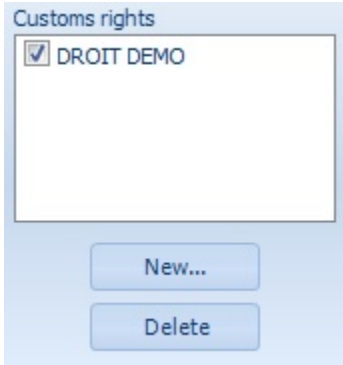
The "**Delete**" button let you remove the selected account (On the left) from the accounts list.

IMPORTANT :

The actual suppression of an account will be definitive once you have clicked on the "Save" button of the ribbon. Once the suppression has been saved, there is no way back.

1.5 - Managing Standard Rights

The "Account access right" panel let you choose the type of rights linked to a user account.

You have the possibility to choose presets families of rights in Winpass or create your own.

1.5.1 - Predefined Access Rights

When defining a new account, you can choose one or several presets families of rights. These six families have been defined to gather every access rights the most commonly used by standard users of Winpass :
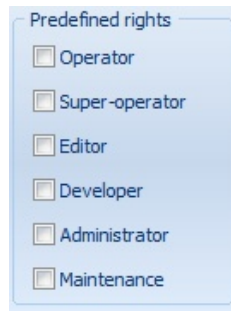
- The "**Operator**" family gathers access rights useful for executing tests on products. This type of rights are a good fit for production operators undertaking tests while not interpreting their results.

- The "**Super-Operator**" family applies to confirmed production operators brought to troubleshoot non-compliant products or prepare products before validating them with Winpass.

- The "**Editor**" family is dedicated to people authorized to modify or update projects or tests documents. This family of rights does not allow the user to test modification made to a project on real products.

- The "**Developer**" family gathers every right required to conceive new projects and tests programs, but also every authorization allowing to debug tests projects on real products.

- The "**Administrator**" family allows the user to access every configuration tool of Winpass 5000 as well as managing files used by it.

- The "**Maintenance**" family is dedicated to users that need to configure Winpass before its usage in production, as well as managers authorized to manage, check and update your testers.


NOTE :

The "Operator" and "Super-Operator" families being complementary, you can select of the two to define an account rights but never both at the same time.

The same goes for "Editor/Developer" and "Administrator/Maintenance" families, which are complementary as well.

Moreover, choosing the "Developer" family make choosing the "Operator/Super-Operator" families superfluous.

1.5.2 - Customized Rights accounts

Managing customized rights allow you to create your own families of rights when you wish to define specific user accounts requiring restrictions of use more secured.
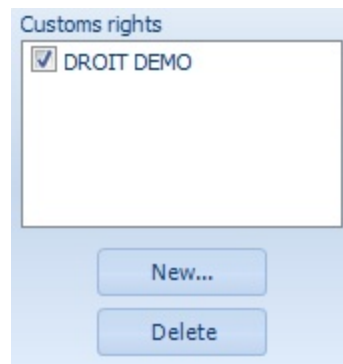
A list displays every customized rights saved and selectable in order to create a new user account.

You can at any time create a new family of customized rights or delete an existing one.

The main contribution of customized rights is to be able to choose groups of rights managed by a family as well as activating or deactivating certain rules that you don't want available for a specific user.

1.5.3 - Families of Rights Setup

Families of rights (Preset or customized) reference groups of rights corresponding to specific modules of Winpass 5000.

These groups of rights allow most of the time to activate or deactivate an access right or a management option peculiar and specific to each group.

A family of rights used for several user accounts is common to every account.

> **IMPORTANT :**
> Activating / Deactivating one or several family rights is effective for the user account that you wish to modify but also for every user account defined with this family of rights.

1.6 - Choosing Groups and Access Rights

The right part of the user account management screen lists every groups and access rights active for every user account you modify or create.

This list allows you to visualize every access rights of an account in particular.

> **NOTE :**
>
> Certain access rights from this list are only visible if specific commercial options of Winpass 5000 have been activated.

1.6.1 - Rights Groups

Rights groups gather access rules specific to each Winpass 5000 module. They are only accessible if they are managed by families of rights chosen for a given user account.

However, when using a customized family of rights for a user account, you can choose one or several groups of rights that have to be managed by this type of family.

The display and the selection of groups of rights is possible by right clicking on the access rights list, which will let appear the complete list of groups managed by the families of rights chosen for the current account.

In that list, and if you are using a customized family of rights, all you have to do is activate or deactivate groups of rights managed by the family you are creating or modifying.

When using preset families of rights, this list does not allow to modify groups of rights managed or not by the current user account.

1.6.2 - Activating Access Rights

Depending on the families of rights chosen for a user account, three cases can appear for each access right or authorization from the list :

- ✔ : A green tick indicates that the access right is active and customizable.

- ✗ : A red cross indicates the usage ban of a tool or the non-management of an option. You can activate this tool by clicking on it.

- ✔ / ✗ : A gray tick or a cross indicates that you cannot modify this right management, its operating method is specific to the family of rights of the account.

1.7 - Note : User's Rights & Winpass Evolution

The list of available and actionable access rights depends on Winpass 5000's version. This list can be brought to evolve in future versions of the software.

You may need to update your accounts when you wish to use potential new modules integrated to Winpass. It is totally possible that upcoming versions of the software allow you to manage new access rights and/or new groups of rights.

In every case, existing groups of rights and access rights will never be questioned and their impact on Winpass' operation will be identical from one version to another.

2 - Testers Management

Winpass 5000 is conceived to handle and use wire testers from the Synor 5000 range. Its goal is to pilot these testers in order to undertake every test required to validate your products connected to them.

That is why Winpass is first and foremost considered as a HCI (Human-Computer interface) dedicated to managing wire testers. Therefore, you will only be able to exploit every function of this software while using it linked to a tester.

Winpass 5000 makes available every tool allowing to manage connections and dialog between your testers and your users.

It is also capable of differentiating testers configurations and exploit every functions specific to a tester. You can also undertake operations of maintenance, update and integrity check on your devices.

Finally, Winpass 5000 offers the possibility to emulate testers configurations; This "virtual" operating mode gives you access to every function of a tester, aiming at conceiving dedicated tests projects, freeing you from a "Physical" link with a tester.

2.1 - Principles / Operating Rules

2.1.1 - Physical Tester Structure

Un testeur est un ensemble de cartes de mesures, de commutation, ... rangées dans un ou plusieurs coffrets. Ces coffrets forment l'architecture du testeur.

A chaque connexion de Winpass 5000 au testeur, son architecture est récupérée et s'affiche sous forme d'une hiérarchie :
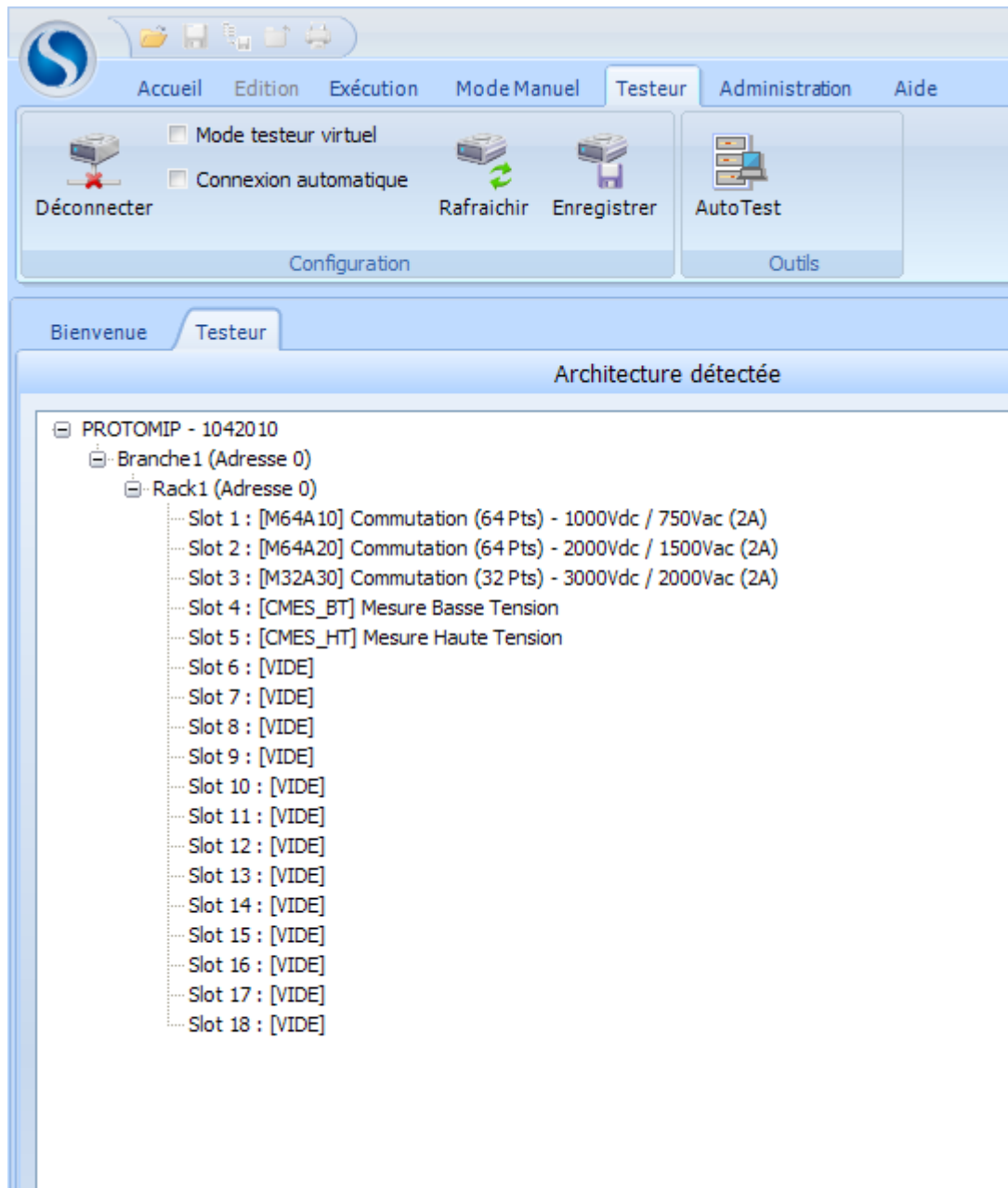
Tout en haut de la hiérarchie, on retrouve le nom du testeur avec son numéro de série. La ou les branches, qui forment la ramification principale, constituent les niveau du dessous. Chacune de ces branches accueille un ou plusieurs racks, ou coffrets. Les différentes cartes sont rangés dans chacun des racks qui composent le testeur.

Voir aussi : gestion des testeurs, informations stockées 142, configurations sauvegardées 143, définition d'un testeur virtuel, connexion/déconnexion.

2.1.2 - Monitoring Options

2.1.3 - Notions about Communicating with Testers
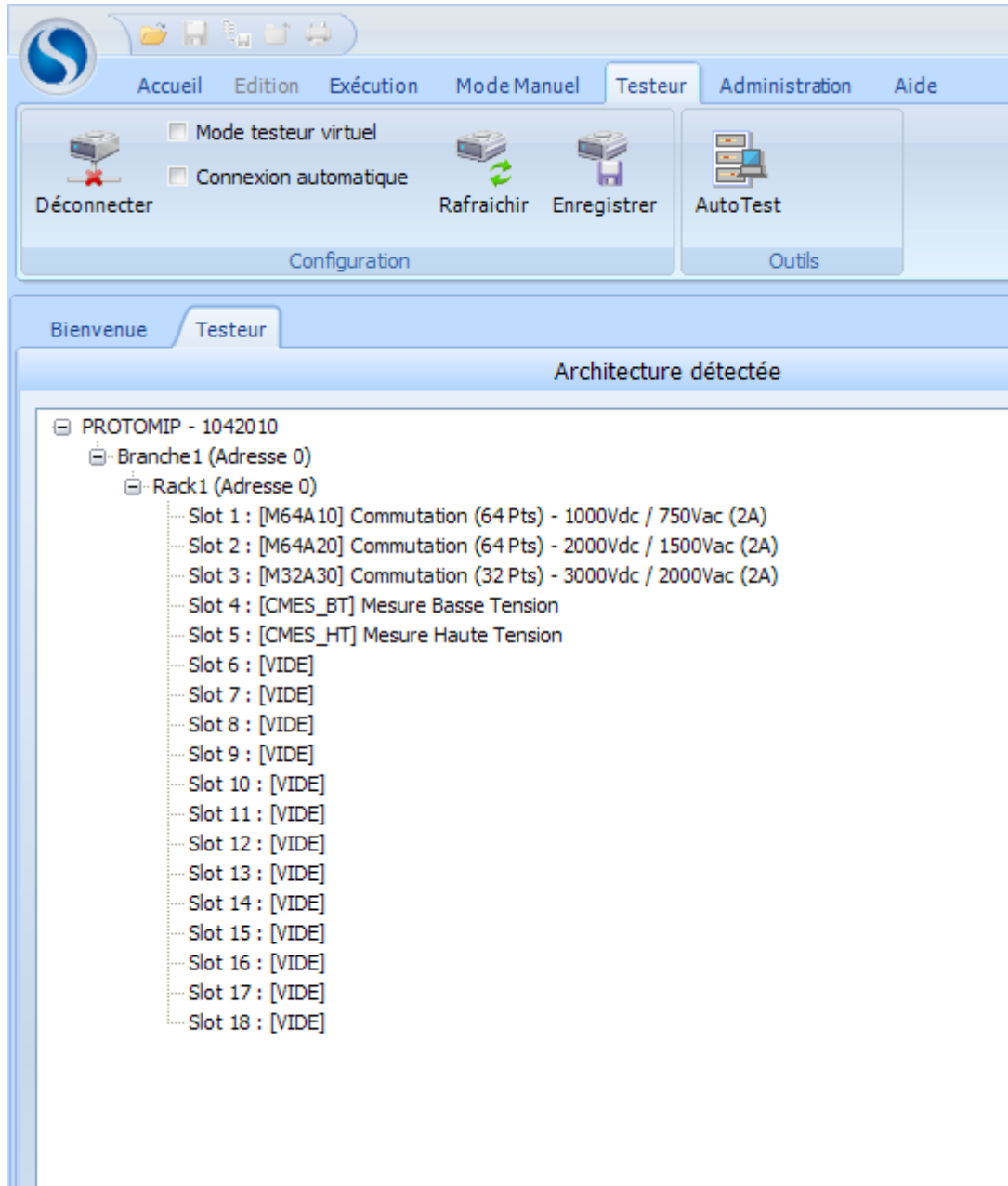
2.2 - Context / Authorization

2.2.1 - Software Configuration

Une fois que la connexion entre le logiciel et le testeur est faite, Winpass 5000 vient lire la configuration du testeur.

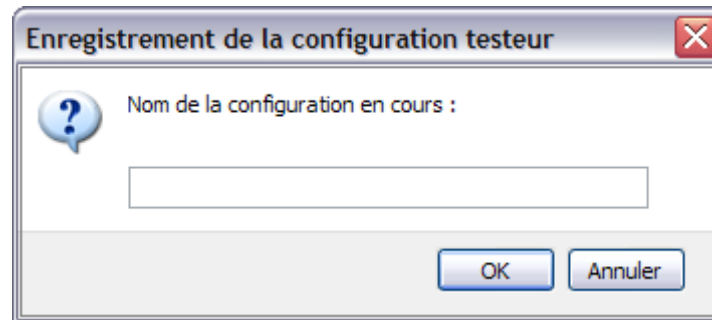Celle-ci se compose de branches, de racks et de cartes.
Chaque branche peut recevoir plusieurs racks, et un rack peut accueillir au maximum 18 cartes.

L'architecture s'affiche sous la forme d'une arborescence. Le premier élément de cette arborescence est le nom du testeur, avec son numéro de série. La ou les branches détectées s'affichent au niveau directement inférieur, vient ensuite le ou les racks de chaque branche. Pour chaque rack reconnu, les différentes cartes qui y sont rangées apparaissent.

L'architecture du testeur utilisé ici se compose d'une branche, d'un rack, de 3 cartes de commutation et de 2 cartes de mesure.

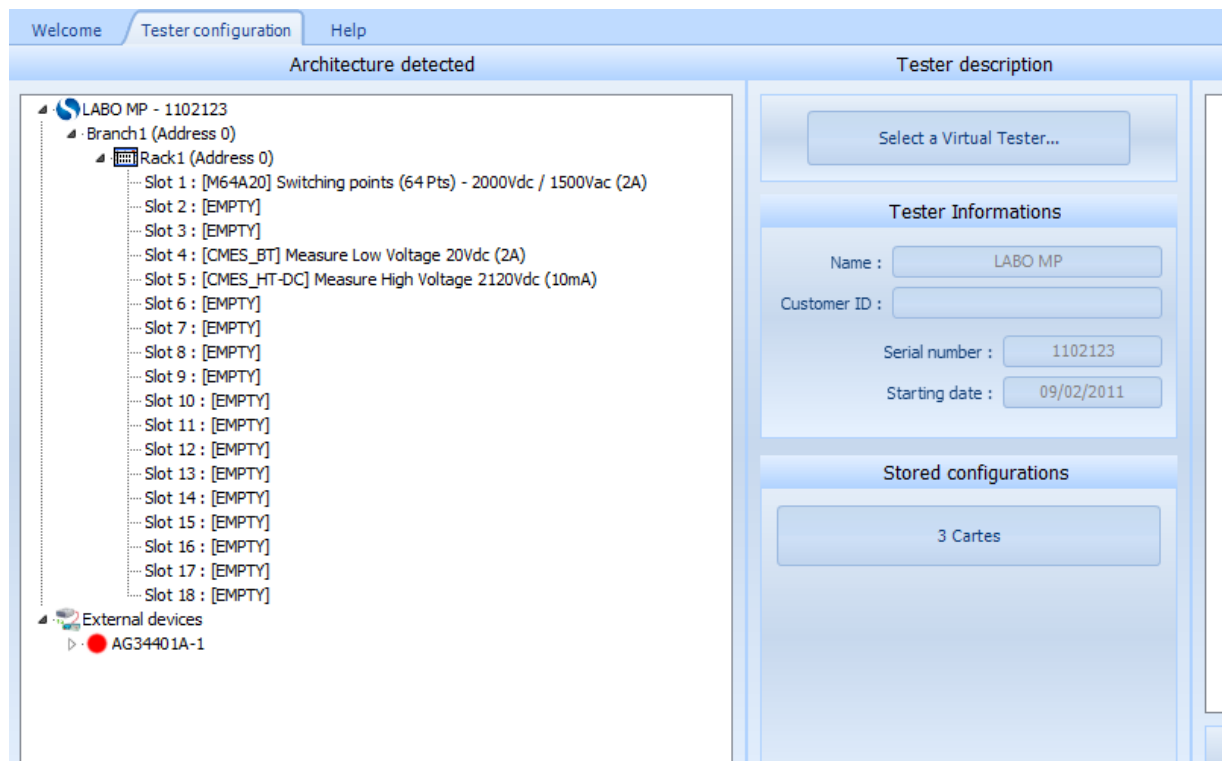Si cette architecture n'avait jamais été détectée auparavant par le logiciel (premier lancement du logiciel après installation, ajout/suppression d'éléments sur l'architecture du testeur, ...), une fenêtre s'affiche pour donner un nom à cette configuration :
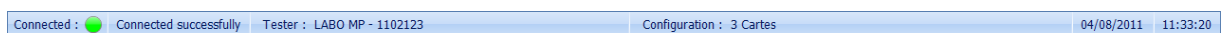
Enregistrement de la configuration testeur

? Nom de la configuration en cours :

OK   Annuler

Cette information apparaît d'ailleurs dans la barre d'état du logiciel :

Connected : 🟢 Connected successfully   Tester : LABO MP - 1102123 | Configuration : UNKNOWN | 04/08/2011 | 11:30:27
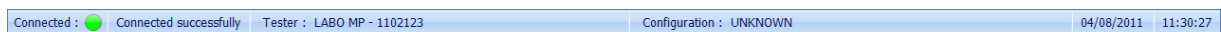
Une fois renseignée, la nouvelle configuration apparaît dans le logiciel, au centre de l'écran :

Welcome | Tester configuration | Help

Architecture detected | **Tester description**

◢ LABO MP - 1102123
  ◢ Branch1 (Address 0)
    ◢ Rack1 (Address 0)
      Slot 1 : [M64A20] Switching points (64 Pts) - 2000Vdc / 1500Vac (2A)
      Slot 2 : [EMPTY]
      Slot 3 : [EMPTY]
      Slot 4 : [CMES_BT] Measure Low Voltage 20Vdc (2A)
      Slot 5 : [CMES_HT-DC] Measure High Voltage 2120Vdc (10mA)
      Slot 6 : [EMPTY]
      Slot 7 : [EMPTY]
      Slot 8 : [EMPTY]
      Slot 9 : [EMPTY]
      Slot 10 : [EMPTY]
      Slot 11 : [EMPTY]
      Slot 12 : [EMPTY]
      Slot 13 : [EMPTY]
      Slot 14 : [EMPTY]
      Slot 15 : [EMPTY]
      Slot 16 : [EMPTY]
      Slot 17 : [EMPTY]
      Slot 18 : [EMPTY]
  ◢ External devices
    ▷ 🔴 AG34401A-1

Select a Virtual Tester...

Tester Informations

Name : LABO MP
Customer ID :
Serial number : 1102123
Starting date : 09/02/2011

Stored configurations

3 Cartes

La barre d'état est ensuite actualisée :

Connected : 🟢 Connected successfully   Tester : LABO MP - 1102123 | Configuration : 3 Cartes | 04/08/2011 | 11:33:20

Voir aussi :

2.2.2 - Users rights (Session)

2.3 - Connecting / Disconnecting from a Tester

2.3.1 - Informations about the Establishment of a Connection

2.3.2 - Virtual / Physical Mode

Winpass 5000 permet de fonctionner en mode "hors-connexion", ou mode "virtuel".

En effet, la connexion peut se faire sur un testeur virtuel, en chargeant une configuration au préalable enregistrée dans la liste.

Il est donc possible de travailler avec le logiciel sans que le testeur ne soit raccordé à l'ordinateur équipé de Winpass 5000.

Ce mode de fonctionnement permet également d'élaborer des projets de tests, en correspondance avec l'architecture du testeur chargée.
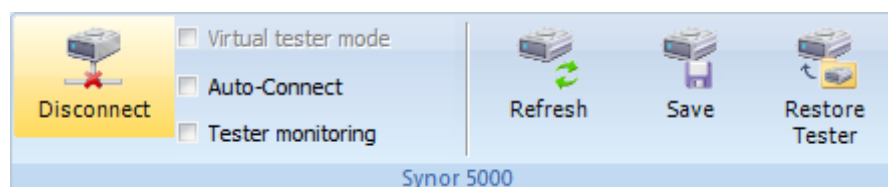
Pour se connecter à un testeur virtuel, il faut cocher la case correspondante et cliquer sur le bouton de connexion, dans l'onglet "Testeur" :
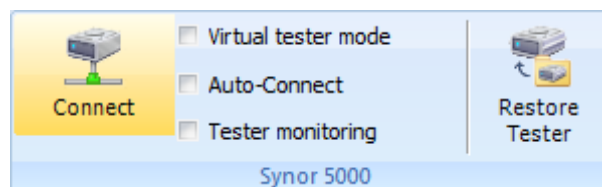
L'architecture correspondant à la configuration se charge dans l'arborescence de droite :
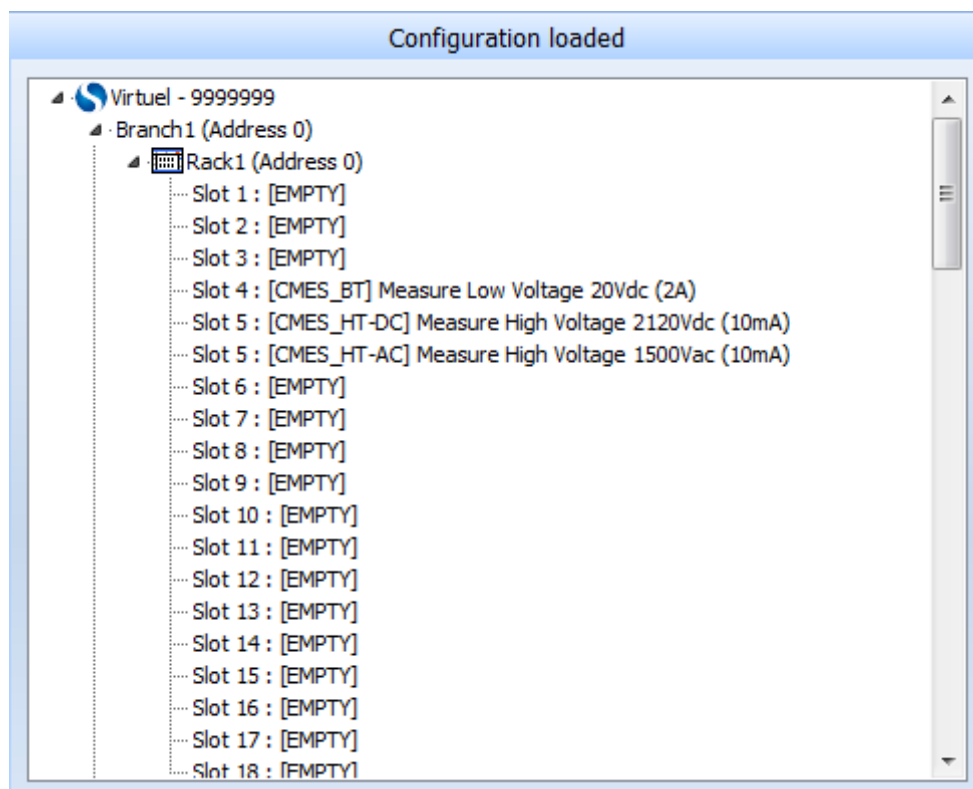
La déconnexion du testeur virtuel se fait en cliquant sur le même bouton :

> **Remarque : après que le connexion soit établie, la LED de la barre d'état change de couleur. Elle passe du rouge au jaune, pour rappeler le mode de travail "hors-connexion".**

Voir aussi : la barre d'état 42, la connexion/déconnexion 51.

2.3.3 - Automatic Mode

2.3.4 - Monitoring / Detecting a Tester

2.4 - Testers Configuration

2.4.1 - Identifying Testers

Dès que Winpass 5000 se connecte à un testeur, le logiciel récupère différentes informations, comme le nom du testeur, le code client, le numéro de série du testeur et sa date de mise en service.

Les deux premières informations peuvent être modifiées (en cliquant sur le bouton ci-dessous), alors que les deux dernière ne le peuvent pas :

Une fenêtre permet de saisir le nouveau nom du testeur :

> **Remarque : la modification du code client se fait de la même manière, en cliquant sur le bouton correspondant.**

> **Remarque : les modifications apportées sont stockées sur le testeur et restaurées aux prochaines connexions.**

Voir aussi : gestions des testeurs, architecture ⌐137⌐, configurations sauvegardées ⌐143⌐, définition d'un testeur virtuel, connexion/déconnexion.

2.4.2 - Handled External Devices

2.4.3 - Saving Infos about a Tester Configuration

Après la première lecture de l'architecture du testeur, Winpass 5000 propose de sauvegarder la configuration qu'il a détecté :

La configuration sauvegardée permet d'être utilisée par la suite en mode testeur virtuel par le logiciel pour la création de projets de tests.

> **Remarque : dès qu'un configuration inconnue aura été détectée (après l'ajout d'une carte, ...), Winpass 5000 proposera systématiquement de sauvegarder la nouvelle configuration.**

Voir aussi : gestions des testeurs, architecture ⌐137⌐, informations stockées ⌐142⌐, définition d'un testeur virtuel, connexion/déconnexion.

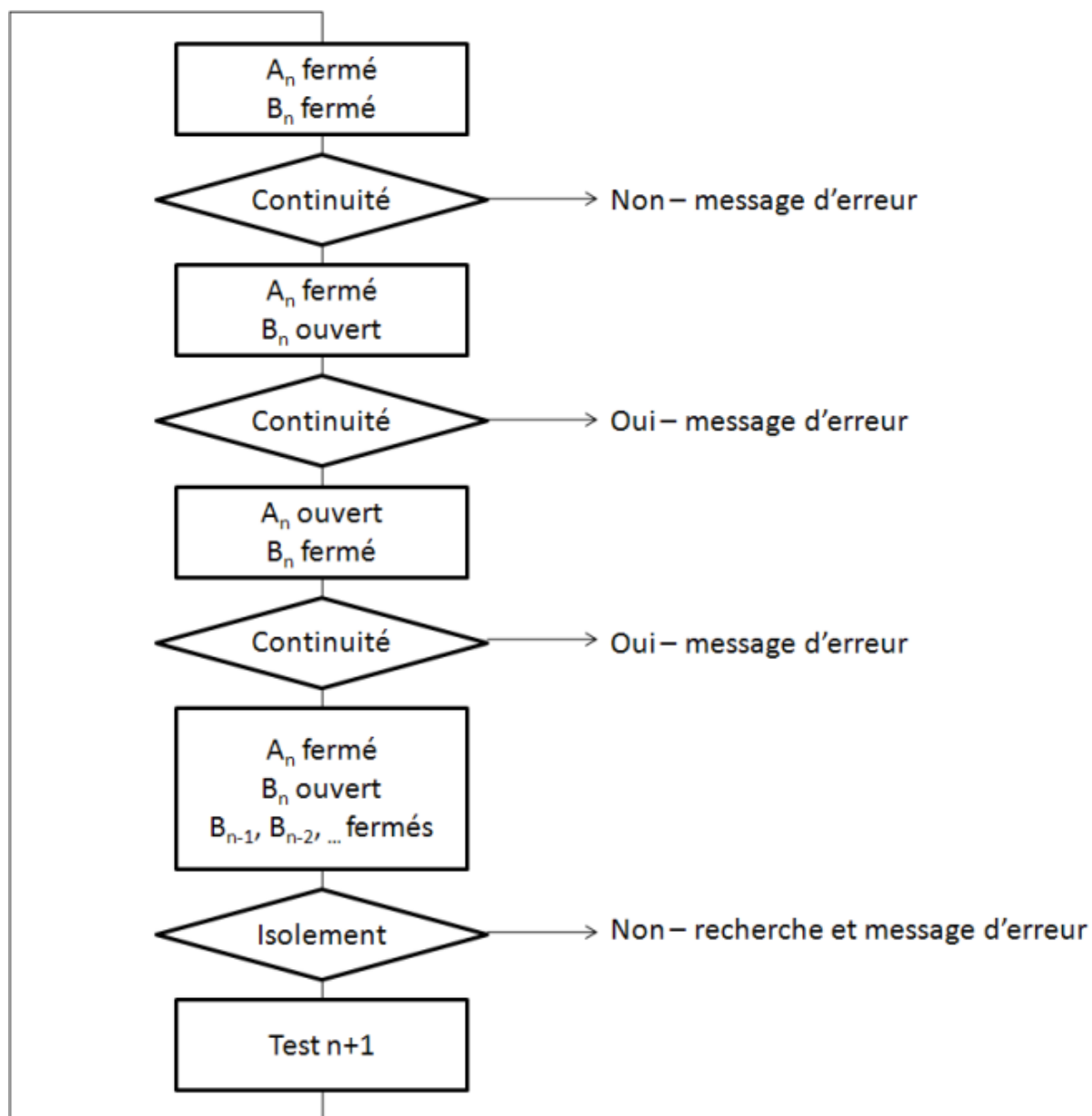## 2.5 - Managing Virtual Testers

2.5.1 - Portability / Re-Usage

2.5.2 - Simulation / Limitations

2.6 - Selftest

Le testeur est utilisé pour contrôler la qualité d'un câblage issu d'une production.
Les résultats d'un test ne peuvent être corrects que si l'état du testeur est irréprochable. Pour cela, un autotest de l'appareil permet de certifier l'état des différents relais qui composent le testeur.
Le testeur possède 3 moyens d'autotest. Les autotests automatiques, les autotests à la demande et les autotests programmés.

Pour chaque point testeur, les deux relais qui le composent son testés en ouverture et en fermeture (quatre tests de continuité). Ensuite, le point est testé en isolement par rapport aux points inférieurs, comme l'illustre le diagramme suivant :

| A$_n$ fermé B$_n$ fermé |
| Continuité → Non – message d'erreur |
| A$_n$ fermé B$_n$ ouvert |
| Continuité → Oui – message d'erreur |
| A$_n$ ouvert B$_n$ fermé |
| Continuité → Oui – message d'erreur |
| A$_n$ fermé B$_n$ ouvert B$_{n-1}$, B$_{n-2}$, ... fermés |
| Isolement → Non – recherche et message d'erreur |
| Test n+1 |

Remarque : un autotest ne peut être réalisé que si un testeur est connecté à l'appareil.

Pour exécuter un autotest, il faut cliquer sur le bouton suivant, dans le menu "Testeur" :
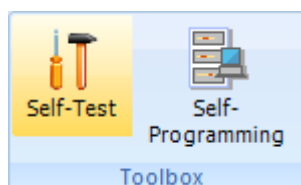
La fenêtre de l'autotest apparaît ; les paramètres de continuité et d'isolement se règlent dans la partie gauche.

Au centre figurent les résultats d'exécution de l'autotest et les options d'exécution se règlent sur la droite (test du système, de la partie commutation et de l'interface quatre fils).

L'exécution débute après avoir cliqué sur ce bouton :

<div style="border: 1px solid #000; padding: 8px; text-align: center;">**Remarque : le premier et le dernier point testeur sont affichés par défaut dans la fenêtre.**</div>

Voir aussi : outils testeur, pointe de touche, auto-programmation, mode autonome.

2.6.1 - Configuration / Properties

2.6.2 - Results Analysis

2.7 - Updating & Reseting Testers

2.7.1 - Firmware Management

2.7.2 - Managing Testers' Images

Eaton is dedicated to ensuring that reliable, efficient and safe power is available when it's needed most. With unparalleled knowledge of electrical power managment across industries, experts at eaton deliver customized, integrated solutions to solve our customers most critical challenges.

Our focus is on delivering the right solution for application. But, decision makers demand more than just innovative products. They turn to eaton for an unwavering commitment to personal support that makes customer success a top priority.

For move information, visit www.eaton.eu/electrical

To contact a Sefelec salesperson or local

distributor, please visit

www.sefelec.com

Eaton
Sefelec sas
19 rue des Campanules
F-77185 Lognes
France

Siège social
+33 (0)1 64 11 83 40

Agence Sud Est
+33 (0)4 74 60 79 78

Agence Sud Ouest
+33 (0)5 56 80 53 22

Service Après Vente
+33 (0)1 64 11 83 48

Fax
+33 (0)1 60 17 35 01

Export Sales
+33 (0)1 64 11 83 42

Export Fax
+33 (0)1 60 17 35 01

Export Service
+33 (0)1 64 11 83 48

Eaton
Sefelec GmbH
Karl-Bold-Str. 40
D-77855 Achern
Deutschland

Service-Abteilung
+49 (0) 78416407717

Fax
+49 (0)78416407729

Eaton
Semelec sas (Our Calibration Division)
11 avenue de l'Atlantique - Les Ulis
F-91955 Courtaboeuf Cedex
France

Siège social
+33 (0)1 69 07 64 58

Fax
+33 (0)1 64 46 41 20

Eaton Industries Manufacturing GmbH
Electrical Sector EMEA
Route de la Longeraie
71110 Morges, Switzerland
Eaton.eu

E·AT·N

Powering Business Worldwide